



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**SISTEMA DE INFORMACIÓN
CON DISPOSITIVOS PORTÁTILES
INALÁMBRICOS USANDO
SERVICIOS WEB**

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

PRESENTAN:

SAUL CASTRO OLVERA

GERARDO M. ORTIZ BANDERA



DIRECTOR DE TESIS:

ING. ALBERTO GONZÁLEZ GUIZAR



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos:

A mis abuelitas Ana María[†] y María del Carmen[†]. Por todo su cariño, apoyo y cuidados incondicionales. La promesa cumplida y una muy especial y cariñosa dedicatoria a su memoria.

A mis abuelos Eduardo y Gonzalo. Por su cariño, atención y por mostrarme la entereza de seguir adelante a pesar de las adversidades.

A mis padres José Luis y María del Carmen. Por brindarme todos sus desvelos, sus esfuerzos, su cariño, su paciencia y sus palabras de aliento.

A mis hermanos Jocelyn y Nestor. Porque juntos hemos aprendido la importancia de la unidad de la familia.

A mi familia. Por enseñarme el valor de los frutos del propio esfuerzo y a ser mejor persona.

A Alejandra. Por el apoyo y aliento incondicional, por la oportunidad de compartir nuestras vidas, por toda la ternura, cariño, amor y momentos felices que me has regalado. Je t'aime beaucoup.

A Lorena. Por la amistad sin la cual nunca hubiera llegado a donde estoy. Gracias por estar cerca siempre. Te Quiero Mucho.

A Germán y Beto. que me han permitido aprender de su experiencia y ejemplo, y me han dado armas para salir siempre adelante. Gracias por todas las enseñanzas y oportunidades que me han brindado.

A todos mis amigos. De primaria (Carlos, Jorge, Ramón, Álvaro), secundaria (Rubén, Oscar), preparatoria (Víctor, Ricardo, Edith, Abigail, José Juan, Alain Daniel, Sergio[†], Juan A.), de la facultad (Abraham, Ariadna, Carmen) y los de AAA (Marcela, Gerardo, Juan Carlos, Jorge, César, Germán, Beto, Raymundo, Ismael), y a todos los que me hacen falta, porque gracias a ellos descubrí el verdadero valor de la amistad. Gracias en especial a Gerardo por la oportunidad de ser compañeros en esta aventura que finalmente ha culminado.

A la ENP #4, a la Facultad de Ingeniería y a la UNAM. Porque me han dado lo único que en verdad nadie puede quitarme. Por todo el conocimiento, las experiencias, las alegrías y las tristezas vividas dentro de ellas. Por darme la oportunidad de llamar a la UNAM orgullosamente "MI alma mater". Gracias también a todos los maestros, porque la simple intención de querer enseñar a otros los hace dignos de mi admiración.

Saúl

Agradecimientos:

A mis padres Mateo y Lupita.

Por que este logro es gracias a su apoyo, esfuerzo y cariño. Por enseñarme a luchar por mis sueños y exigirme a ser mejor día con día, y a los que es imposible pagar su cariño como se merecen.

A mis hermanos Lety y Ernesto.

Por su cariño y apoyo cada día. Gracias por su paciencia, compañía y amistad en todo momento. Gracias por soportarme y por que seguimos juntos, por siempre.

Al Sensei Germán

Por darme la oportunidad en un momento difícil, y enseñarme a ser autodidacta.

A Yesica

Por estar conmigo en las buenas y en las malas, por tu amor y tu confianza.

A mi director de tesis, Alberto

Por la asesoría y enseñanza brindada en la realización de esta tesis y proyecto.

A la DGSCA

Gracias por todos estos años de darme la oportunidad de aprender y ser mejor.

A la Universidad Nacional Autónoma de México

El espíritu universitario y el compromiso que significa son un complemento maravilloso a los conocimientos y habilidades adquiridos bajo su tutoría.

A mis amigos

Por darme su apoyo incondicional en todo momento, por ayudarme a crecer y darme muchos días felices y llenos de risas, por darme sus hombros para llorar y dejarme entrar a sus vidas. Por compartir conmigo un poquito de cada uno de ustedes.

Gerardo

*"Yo no podría ver tan lejos si no me apoyara en los
hombros de gigantes"*

Isaac Newton (1642-1727)

INDICE

INDICE	1
1. Antecedentes.....	3
1.1. Introducción	5
1.2. Desarrollo de las Telecomunicaciones	10
1.3. Globalización y telecomunicaciones	12
1.4. El porqué de "a quien sea, a cualquier hora, en cualquier lugar, en cualquier dispositivo"	19
2. Conceptos Básicos	21
2.1. Introducción	23
2.2. Historia de las computadoras	24
2.2.1. Historia de las Computadoras como Sistemas de Información	24
2.3. Internet y los sistemas de información	31
2.3.1. Internet hasta finales del Siglo XX	31
2.3.2. Internet en el Siglo XXI	33
2.3.3. Internet 2	34
2.3.4. Internet Siempre Disponible	35
2.4. Nuevas Tecnologías	36
2.4.1. Dispositivos Portátiles (Celulares, PDA's, etc.)	36
2.4.2. Sistemas de Comunicaciones Digitales	43
2.4.3. Necesidades y tendencias en la actualidad. Sistemas de Información Inalámbricos	51
2.5. Protocolos de comunicación e intercambio de información	53
2.6. El modelo de referencia OSI	61
2.7. Java	64
2.7.1. Historia de java	64
2.7.2. Aspectos Técnicos de Java	68
2.7.3. J2EE	74
2.7.4. J2ME	78
3. Análisis	79
3.1. Nuevas Tecnologías Para la Integración de Sistemas de Información	81
3.1.1. Estandarización	81
3.1.2. Arquitecturas de Sistemas de Información con Servicios Web	85
3.1.3. Seguridad y Servicios Web	91
3.1.4. Arquitectura SunONE de Sun Microsystems	94
3.1.5. Arquitectura Microsoft .NET	99

3.2. Justificación: J2EE (SunONE) vs. Microsoft .NET	103
3.2.1. La visión de J2EE	105
3.2.2. La visión de Microsoft .NET	107
3.3. La Plataforma J2ME para Dispositivos Portátiles	119
3.4. Wireless: Un Mundo sin Cables	124
3.4.1. WLAN's y Tecnologías de Comunicación Inalámbricas	124
3.4.2. Estándares de Comunicaciones Inalámbricas	133
3.4.3. Java Inalámbrico.	140
4. Desarrollo	149
4.1. Antecedentes	151
4.2. Solución propuesta	154
4.3. Implementación	156
4.3.1. Sistema SIR	156
4.3.2. Escenario	162
4.3.3. Arquitectura	171
4.3.4. Ventajas y Costos	174
5. Conclusiones	181
5.1. Redes inalámbricas	183
5.2. Servicios Web	185
5.3. J2ME	186
5.4. Conclusiones Finales	188
6. Anexos	189
* GLOSARIO	191
* BIBLIOGRAFÍA	211
* ENLACES DE INTERÉS	213

1.

Antecedentes

1.1. Introducción

La palabra *comunicación* pertenece al lenguaje cotidiano; se usa y se conoce su significado en forma intuitiva, nadie subestima su importancia, pero pocas personas podrían definirla en forma precisa.

Desde el punto de vista etimológico, la palabra *comunicación* proviene de la raíz latina *communicare*, que significa hacer de conocimiento común algún tipo de información. Otra palabra, de un uso también muy común en la actualidad, es *telecomunicaciones*, que proviene de la composición de comunicación y el prefijo *tele* (lejos); *telecomunicaciones* significa entonces comunicar a distancia.

En una de las obras de mayor repercusión sobre las telecomunicaciones modernas, "*A Mathematical Theory of Communication*", de C. E. Shannon y W. Weaver, editada por la Universidad de Illinois en 1949, se define el concepto de comunicación de una manera muy sencilla: "comunicación son todos aquellos procedimientos por medio de los cuales una mente afecta a otra". Esto incluye voz, texto impreso o escrito, música, artes, teatro y danza. En la misma obra se amplía la idea anterior para incluir la posibilidad de comunicación entre máquinas: "comunicación son todos aquellos procedimientos por medio de los cuales un mecanismo afecta la operación de otro".

La información, como entidad susceptible de ser comunicada, es coleccionable, almacenable y/o reproducible. Se utiliza para tomar decisiones, conduce también a conclusiones acertadas o equivocadas, puesto que puede ser interpretada de diversas formas por distintos individuos, dependiendo de muchos factores subjetivos y del contexto en que se encuentre la persona que la recibe e interpreta. Así como es posible comunicar una noticia, también se comunican los estados de ánimo, opiniones o conocimientos.

Citamos un caso a manera de ejemplo: el 19 de junio de 1815, en la Bolsa de Valores de Londres, un mensajero proveniente de Ostend, Bélgica, entregó en secreto una noticia a Nathan Rothschild. De inmediato, Rothschild vendió todas sus acciones. Los observadores, enterados de que Rothschild tenía fuentes confiables de información, lo imitaron porque supusieron que ello se debía a una victoria napoleónica en Waterloo, lo cual pondría en serios problemas a la prosperidad británica y su hegemonía sobre Europa. Hacia el mediodía, en un mercado de valores totalmente deprimido, Rothschild compró nuevamente todos los valores que él y todos los que lo imitaron habían vendido, a sólo una fracción del precio de las ventas originales. Horas más tarde llegó la noticia de la victoria de Wellington, con lo cual los valores no sólo recuperaron su precio de la mañana, sino que, al estar en manos de una sola persona, éste aumentó... y con ello Rothschild ganó una fortuna en pocas horas, por tener y manejar adecuadamente información que nadie más poseía.

La mayor influencia sobre las comunicaciones la tuvo la Segunda Guerra Mundial: en esa época la humanidad ya se encontraba en la frontera de la revolución tecnológica, misma que las actuales generaciones hemos tenido la oportunidad de presenciar desde hace algunos años. Muchos de los sucesos que condujeron a la conclusión de la guerra, con el resultado que todos conocemos, estuvieron relacionados con la disponibilidad de información oportuna o con la intercepción ingeniosa de información del enemigo.

Antecedentes

Los requerimientos de comunicaciones instantáneas, seguras y privadas de esa época fueron determinantes para que las comunicaciones sean lo que son hoy en día. Se sabe de muchos escándalos financieros en los cuales las personas que poseen información confidencial antes que otras, la usan a su favor, y ganan grandes capitales (este uso personal de información confidencial es ilegal en muchos países).

En estos días es difícil pensar que alguien niegue conscientemente que la información tiene un valor; la información ha ido ganando importancia conforme la gente que toma decisiones se convence de que ésta se puede asociar a un valor real, frecuentemente ligado a un valor material o económico. Esto es distinto de lo que ocurría en otras épocas, en que predominaban otros bienes y servicios, que tenían mayor valor económico. A las épocas de grandes cambios en la historia de la humanidad, se les han asignado nombres especiales: el Renacimiento, la Ilustración, Revolución Industrial, etc. En nuestros días, primer década del siglo XXI, es de tal importancia poseer, administrar y transmitir información, que toda la humanidad se ve y se seguirá viendo afectada, influida y posiblemente dominada por quienes tienen, administran y transmiten este recurso, razón por la cual a esta época se le han impuesto los calificativos de *sociedad de la información* o de *revolución electrónica*, éste último debido a la facilidad con que se transmite la información por medio de los sistemas modernos basados en dispositivos electrónicos.

Uno de los aspectos más abstractos e importantes de la información es el valor. La información tiene su valor para una determinada persona, en términos de la utilidad que puede representarle para la toma de la mejor decisión en el momento en que sea preciso (en la actualidad, la mejor decisión generalmente implica una conveniencia, una ganancia económica o un ahorro en los costos). El valor de la información puede disminuir a lo largo del tiempo. Es decir, en un momento determinado a alguien le puede interesar contar con cierta información, pero ese interés puede decrecer o incluso desaparecer algún tiempo después (por ejemplo: difícilmente a un inversionista en una casa de bolsa podría interesarle información bursátil del día de ayer, o inclusive de hace un par de horas). Por otra parte, es necesario que la información sea de interés para el individuo que la adquiere o recibe, quien, además, no debe conocer *a priori* su contenido; en caso contrario, dicha información le resultará irrelevante. Es evidente que este estado de incertidumbre no necesariamente tiene que ser consciente ni voluntario.

La información se origina en una fuente y se hace llegar a su destinatario por medio de un mensaje a través de un canal de comunicación; el destinatario generalmente se encuentra en un punto geográfico distante, o por lo menos, separado de la fuente. La distancia entre fuente y destinatario puede variar desde pocos centímetros (al hablar frente a frente a un volumen normal) hasta cientos y aun miles de kilómetros (como es el caso de transmisiones telefónicas intercontinentales o de transmisiones desde y hacia naves espaciales). Esto constituye precisamente el problema central de las telecomunicaciones, ya que al haber una fuente que genera información en un punto y un destinatario en otro punto geográfico distante del primero, se trata de saber cuál es la mejor manera de hacer llegar al destinatario la información generada por la fuente, de manera rápida (por la dependencia temporal de la importancia de la información), segura (para garantizar que la información no caiga en manos de alguien que haga mal uso de ella, o a quien simplemente no estaba destinada), y veraz (para garantizar que en el proceso de transmisión no se alteró el contenido de la información).

En nuestros días, influidos fuertemente por aspectos de tipo económico, intervienen además otros factores, tales como el costo de hacer llegar la información de la fuente a su destino.

El problema central de las telecomunicaciones también fue definido con claridad por Shannon, nuevamente con una sencillez asombrosa, quien estableció que un sistema de comunicaciones consiste en cinco componentes:

1. Una fuente de información.
2. Un transmisor de información cuya función consiste en depositar la información proveniente de la fuente en un canal de comunicaciones.
3. Un canal de comunicaciones, a través del cual se hace llegar la información del emisor al receptor.
4. Un receptor que realiza las funciones inversas del transmisor, es decir, extrae la información del canal y la entrega al destinatario.
5. Un destinatario.

Un mensaje se usa para hacer llegar información de fuente a destino, y no es lo mismo un mensaje que la información que éste contiene. Considérese el siguiente ejemplo: Una persona (A) desea enviar cierta cantidad de dinero por medio de un giro telegráfico a otra persona (B). En este caso, A es la fuente, B el destinatario. La información es aquello necesario para conocer la cantidad de dinero y para originar la entrega del mismo a B, y el mensaje es el conjunto de palabras o símbolos telegráficos necesarios para que B conozca la intención de A y para que B pueda disponer del dinero que A le envía.

Desde la Antigüedad se reconocía la necesidad de transmitir información a distancia. Desde entonces, las soluciones a este problema han estado íntimamente relacionadas con el desarrollo cultural, social y político de la humanidad. Para transmitir información entre dos puntos, primero debe ser *envasada* en un *contenedor*, que posteriormente se enviará a través de un *canal*; dicho proceso es tan abstracto como el de la misma información, pero se explica con la ayuda de algunos ejemplos: si la información consiste en ideas, decisiones o estados de ánimo, las maneras de enviarla a distancia es por medio de palabras, texto impreso, imágenes, ondas electromagnéticas o señales intermitentes de humo, por mencionar sólo algunas, y los canales de comunicación para cada uno de ellos son respectivamente el aire, el correo, un cable de televisión y la atmósfera, entonces en todos los casos se observa que el medio o canal a través del cual se transmite la información es un elemento que impone restricciones sobre los *contenedores* de la información: una onda acústica generada por una voz sólo puede ser transmitida por un canal que conduzca ondas acústicas y una eléctrica, por medio de un conductor de señales eléctricas. Afortunadamente, hoy en día, con ayuda de la tecnología, es posible solucionar estas limitaciones y convertir señales de un tipo a otro: el precursor de esto es el micrófono, por medio del cual se convierte una señal acústica en eléctrica. El mensaje fue creado por el hombre para comunicarse, es decir, para hacer de conocimiento común algo que en este caso específico es la información.

Antecedentes

Esto es una muestra palpable del ingenio humano: la creación de un mensaje forzosamente implica la necesidad de codificar la información para que sea susceptible de ser enviada o transmitida; no sería posible transmitir una idea sino se utilizara el lenguaje oral, el corporal, el escrito, o algún otro; estos lenguajes son precisamente las versiones codificadas de la información.

Es posible explicar las funciones del codificador de la siguiente manera: así como no se puede enviar una carta (es decir, un sobre de papel que contiene otros papeles en su interior, cuyos símbolos o texto contienen la información que se desea transmitir) a través de un canal telefónico o un satélite, tampoco es posible enviar señales de humo utilizando para ello el correo. Por tanto, es indispensable adaptar el mensaje que contiene la información al canal por el que será transmitido. Ésta es precisamente la función de un codificador.

Para que se complete el proceso de comunicación, se requiere que tanto el que origina el mensaje como el que lo recibe conozcan la forma en que fue codificada la información (esto es, el código que fue empleado); en otras palabras, para que dos personas se comuniquen por la vía oral, es indispensable que ambas hablen el mismo idioma, y para que dos personas se comuniquen por vía telefónica, se requiere que, además de hablar el mismo idioma, ambas tengan a su disposición un aparato telefónico y que ambos estén unidos por medio de conductores de señales.

El hombre, al querer cubrir distancias cada vez mayores, empezó a utilizar sistemas cada vez más complejos, conforme se lo permitían los avances científicos y tecnológicos. Como consecuencia, también comenzó a usar sistemas de codificación tan abstractos como la escritura misma: símbolos basados en señales intermitentes de humo, o en diversas combinaciones de señales de fuego generadas por medio de antorchas. Éstos fueron los precursores de la codificación de la información.

El historiador griego Polibio (204-122 a. de C.) relata que la manera en que se codificaban las 24 letras del alfabeto griego era colocando cada una de ellas en una retícula cuadrada de 5 x 5 unidades: por ejemplo, el código de la letra *alfa*, colocada en el primer espacio, era: primer renglón, primera columna. Se puede afirmar que también fue Polibio quien diseñó el primer sistema digital de comunicaciones sincronizadas. En este caso, se trabajaba en la misma línea visual, de una isla a otra, con dos recipientes cilíndricos de igual tamaño llenos de agua. Ambos tenían un pequeño orificio por donde salía un chorro de agua. Dentro de los recipientes se contaba con una regla que tenía un conjunto de símbolos convencionales: "necesito refuerzos", "necesito alimento", "manden barcos", etc. Por medio de una antorcha se señalizaba (se informaba) de una isla a otra el instante en que debía ser abierto el orificio, y por medio de otra antorcha se señalizaba el instante en que debía ser cerrado. El mensaje transmitido era precisamente aquel que se encontraba a la altura del agua en el momento de cerrar los orificios. Por supuesto que la sincronía era un factor extremadamente crítico; si ésta fallaba el receptor podía interpretar incorrectamente el mensaje.

Es probable que entre los primeros sistemas de los cuales se valió el hombre para transmitir información a distancia estuviera el de los mensajeros humanos. Sin embargo, cuando la distancia era mayor que la que podía recorrer un mensajero (ya sea caminando o cabalgando) en el tiempo requerido para que el destinatario no perdiera interés en la información o para que ésta llegara a tiempo para la toma de decisiones, surgió el sistema denominado de *relevos*.

Esta nueva evidencia del ingenio humano está documentada en fuentes históricas sobre las comunicaciones en el Imperio Romano: ahí se menciona la existencia de *mutaciones* y *mansiones*, o sea, estaciones para cambio de cabalgadura y para descansar, respectivamente (estos conceptos son precursores de los hoteles y moteles ubicados en las carreteras de hoy).

Un mensajero que en aquella época tenía que recorrer largas distancias estaba también forzado a salvar todas las asperezas topográficas propias de la región. El hombre se percató, entonces, del hecho de que las señales ópticas podían recorrer mayores distancias y más rápidamente que las señales de tipo acústico. Es decir, una persona puede alcanzar a ver algo que ocurre a una distancia mayor que aquella que puede ser cubierta por medio de sonidos. Esta *nueva tecnología* podía, además, fácilmente salvar obstáculos, como barrancos, cerros, ríos o lagos.

Esquilo (525-456 a. de C.), en su tragedia *Agamenón*, relata cómo alrededor del año 1000 a. de C. ya se utilizaba este sistema en combinación con el de los relevos: se estableció un sistema de comunicación óptica entre Troya y Argos, que contaba con estaciones repetidoras, y lograba cubrir en una noche una distancia de más de 500km (en mar y tierra). Cabe mencionar que en esa época, debido a que las señales ópticas se generaban por medio de antorchas, eran más efectivas las transmisiones durante la noche que durante el día.

En la actualidad persiste el sistema de las repetidoras; la función de un satélite de comunicaciones *estacionario* a 35,000km de distancia de la superficie de la Tierra es la misma que la de un guerrero griego en la cima del monte Atos (el más alto entre Troya y Mikenos), a una altura de 2,033 metros sobre el nivel del mar: recibir información de un punto y retransmitirla a otra sin introducir ninguna modificación.

A pesar de que en el pasado remoto fueron ideados los precursores de las telecomunicaciones modernas, cuando al principio pequeños grupos de individuos y después países enteros construyeron sus propias infraestructuras para satisfacer la necesidad de transmitir información a distancia, los fundamentos técnicos del área datan del pasado y el presente siglo. Una excepción es el correo, que se inició hace ya algunos siglos y en el cual los principios básicos aún perduran: la información que se ha de transmitir se codifica en palabras, que a su vez son plasmadas en papel (o sus predecesores); se utilizaba posteriormente algo similar a lo que ahora se conoce como un sobre, el cual era depositado en un buzón (o su equivalente). Todos los sobres depositados en los buzones eran recolectados y transportados a una oficina central en donde se seleccionaba la ruta que había de seguir cada sobre; para llegar a su destino se pasaba por una etapa de transporte que incluía todos los recursos disponibles en cada época, hasta que finalmente se entregaba al destinatario. Durante un largo periodo en la historia de la humanidad, ésta fue la única forma de comunicación a distancia, desde luego adaptándose a las posibilidades que iban ofreciendo los nuevos adelantos tecnológicos: en lo que se refiere al transporte, del caballo se pasó a los barcos y los ferrocarriles, después a los automóviles y por último a los aviones.

1.2. Desarrollo de las Telecomunicaciones

Hablar en detalle del desarrollo de las telecomunicaciones modernas, desde su origen histórico, hasta las más recientes aplicaciones, es una enorme tarea, y un tema que sale del alcance de este documento, de forma que lo que a continuación se presenta, es un resumen a modo de contexto histórico, del desarrollo de las telecomunicaciones.

Siglo XVIII: Los principios

Las bases que dieron lugar al desarrollo de las primeras formas de tecnologías de la información, son en cierta forma conocidas desde la antigüedad (los griegos conocían el fenómeno del electromagnetismo, por ejemplo), sin embargo, la investigación profunda y científica de estos principios básicos, se comenzó a dar durante el siglo XVIII con los pineros de la física, como son: A. Volta (1745-1827), G. S. Ohm (1787-1854), J. C. Maxwell (1831-1879), A. M. Ampere (1775-1836), J. Henry (1797-1878), M. Faraday (1791-1867), H. C. Oersted (1777-1851), C. Wheatstone (1802-1875), K. F. Gauss (1777-1855), por nombrar algunos. Los primeros sistemas fueron concebidos y construidos por C. Chappe durante la segunda mitad del S. XVIII, quien además utilizó por primera vez la palabra *telégrafo* para su aparato que servía para enviar mensajes a distancia.

Siglo XIX: La aplicación

Hacia finales del siglo XVIII y XIX, muchos inventores se dedicaron a encontrar la parte *práctica* (o *la aplicación*) a los descubrimientos científicos hechos durante el siglo XVIII. Se inventa el telégrafo alámbrico en 1844, el inalámbrico en 1874, y el primer teléfono en 1876. Estos inventos, concebidos por personas como S. Morse (1791-1872), G. Marconi (1874-1937), A. G. Bell (1847-1922), H. Hertz (1857-1894), finalmente sientan las bases de lo que es el sistema telefónico que conocemos hoy en día. El teléfono es el sistema más grande, complejo, distribuido y común en el mundo, y también el que tiene mayor influencia socioeconómica, ya que es sencillo de usar, y permite la comunicación casi instantánea entre 2 personas (o sistemas, o computadoras) situadas inclusive en lados opuestos del planeta, todo ello de forma casi automática y prácticamente sin intervención humana.

Mientras las telecomunicaciones comenzaban a ver la luz, hacia finales del siglo, un estadista norteamericano de nombre Herman Hollerith, inventa una máquina electromecánica capaz de procesar información representada en tarjetas perforadas, basado en la idea de los telares de Joseph Marie Jacquard. Esta máquina, era capaz de realizar cálculos numéricos (como lo hacían muchos aparatos mecánicos, ya desde el siglo XVII), pero además, podía llevar conteos, clasificar, y realizar análisis numéricos sobre la información, operaciones que ninguna otra máquina hasta entonces podía realizar.

Los aparatos inventados por Hollerith se usaron por primera vez en el censo poblacional de EU en 1890, y posteriormente para diversas tareas en todo el mundo hasta mitades del siglo XX, representando los inicios de la informática moderna que más tarde se convertiría en uno de los factores más determinantes para desarrollo de las telecomunicaciones.

Siglo XX: El desarrollo

En el S. XX aparecen los sistemas electrónicos, basados primero en relevadores, luego en bulbos, posteriormente en transistores, y finalmente en circuitos integrados como microprocesadores. Se perfecciona el sistema telefónico inventado el siglo pasado, además se comienzan a utilizar sistemas de comunicaciones por medio de ondas electromagnéticas y medios ópticos. Aparecen la radio y la televisión, la telefonía inalámbrica, satelital y celular. Se comienzan a utilizar redes de datos conectadas entre sí (ARPANET), que van creciendo hasta convertirse finalmente en la compleja red de cobertura mundial que hoy conocemos como Internet.

Siglo XXI: Presente y futuro

La complementación mutua de diferentes tecnologías y el desarrollo paralelo y concurrente de muchas de ellas han dado a las telecomunicaciones un grado de avance que hace apenas tres décadas era totalmente insospechable. Se cuenta en estos días con una infraestructura de telecomunicaciones con cobertura global, que ofrece una enorme variedad de sistemas interconectados, y que pone a disposición de los usuarios la más increíble diversidad de servicios de telecomunicaciones. Enumerarlos y saber cómo funcionan constituye todo un reto: desde el servicio básico de telefonía con todas sus modalidades y variaciones, como sus versiones local, de larga distancia, rural, pasando por los distintos esquemas de radiotelefonía, como la móvil y la portátil, hasta llegar al video texto, las redes privadas y públicas de transmisión, así como las redes digitales con servicios integrados, la radiodifusión, la televisión, con sus versiones vía cable, de alta resolución, servicios de valor agregado como el fax, la localización de personas, de vehículos y de flotillas de vehículos en movimiento, y casi todos los servicios que se prestan con las redes modernas de telecomunicaciones (casi todas las palabras que inician con el prefijo tele: telemedicina, teleducación, telecompras, televotaciones, teleconferencias, etc.).

Vale la pena recordar la frase célebre de I. Newton (1642-1727): "yo no podría ver tan lejos si no me apoyara en los hombros de gigantes". Los gigantes de las telecomunicaciones en que se apoyaron los científicos a partir de 1940 fueron las transmisiones radioeléctricas (las cuales permitieron el desarrollo de la televisión, la radio, las microondas y los satélites) y eléctricas (que a su vez dieron origen al teléfono, los cables submarinos y al concepto genérico de redes de telecomunicaciones). Las mentes de los *gigantes* que permitieron llegar a este punto, aparte de que se les clasifique subjetivamente como científicos, ingenieros o tecnólogos, tienen algo en común: fueron mentes geniales y privilegiadas por haber podido poner a disposición de la humanidad sus inventos o desarrollos, sus conocimientos científicos que han cambiado de manera radical el comportamiento del ser humano, dándole la posibilidad de comunicarse casi de forma instantánea, de muy diversas maneras, con casi cualquier otro habitante de este planeta.

Aún existe espacio para seguir explotando sus resultados. Hay que recordar que las ideas geniales provienen de genios, y que, lamentablemente, éstos no abundan, pero también que las ideas verdaderamente innovadoras dan herramientas poderosas y amplias posibilidades de explotación al servicio de la humanidad durante un largo tiempo. Los nuevos gigantes de nuestra época tienen la ventaja de poder apoyarse sobre los hombros y los sólidos resultados de los genios del pasado.

Antecedentes

1.3. Globalización y telecomunicaciones

Las telecomunicaciones modernas son un factor que ha potenciado la globalización. A continuación se mencionarán algunos aspectos históricos sobre el desarrollo de las telecomunicaciones, sus funciones y su influencia cultural, social y económica.

1850: el telégrafo, inicio de las telecomunicaciones

Utilizando la tecnología de la época, con la telegrafía era posible transmitir unas cuantas palabras por minuto a través de algunos kilómetros de distancia. En los siguientes 50 años, con técnicas más complejas, y con base en transmisiones de radio (es decir, inalámbricas) que permitía comunicaciones con velocidades de propagación igual a las que se utilizan hoy en día (a la velocidad de la luz), se logró, en 1870, enviar mensajes a tasas de transmisión del orden de 20 palabras por minuto. En 1901, después de las exitosas transmisiones de Marconi entre Poldhu, Inglaterra, y St. Johns, Canadá, fue posible transmitir, casi independientemente de la distancia entre transmisor y receptor, hasta cientos de palabras por minuto. Pocos avances se verían en las telecomunicaciones durante casi un siglo.

Las telecomunicaciones hasta 1940

El crecimiento y la maduración de las telecomunicaciones, el aumento en su disponibilidad, confiabilidad, seguridad, conectividad y la disminución de los costos reales en los servicios, han sido producto (directa o indirectamente) de avances en diversos campos del conocimiento, como la ingeniería espacial y la aeronáutica, pasando por la ciencia de materiales y física, hasta la tecnología digital; en resumen: el desarrollo de la electrónica y de la computación. Aunque muchos de estos avances han tenido su origen en el uso militar, otros no menos importantes tuvieron sus inicios en aplicaciones civiles, como es el caso del teléfono (cuyos inicios se debieron a la búsqueda de su inventor, Graham Bell, de un sistema que permitiera visualizar las señales de voz y ayudarse en sus labores de enseñanza a personas sordomudas; hasta convertirse en uno de los aparatos de comunicación más utilizados por las sociedades en la actualidad).

Las telecomunicaciones se han convertido en un satisfactor de necesidades cotidianas de un importante número de habitantes y corporaciones de este planeta. Sin embargo, pocos se han preguntado cómo opera cada sistema, y qué importancia tiene en un mundo donde la transmisión de información a distancia es un fenómeno común y cada vez más necesario.

1945: intereses militares, fundamento para el desarrollo de las telecomunicaciones

Los avances que se lograron durante la segunda guerra mundial en el área de las comunicaciones fueron determinantes para su desarrollo. En aquellos años se llevó al extremo el ingenio humano, al diseñar sistemas más rápidos, seguros, y privados que los conocidos hasta ese momento. Frecuentemente se implantaban soluciones un tanto empíricas, sin tener aún dominados todos los aspectos científicos que se requerían para ampliar los conocimientos del área. Importaba en especial un aspecto: cómo emplear las comunicaciones para beneficiar los intereses militares que dominaban en aquellos días.

Las dos guerras mundiales fueron acontecimientos que en mayor medida han afectado no únicamente a las telecomunicaciones modernas, sino a la tecnología y la investigación científica en general (un ejemplo de esto es el desarrollo de los aviones). Ambas guerras fueron las responsables de convertir experimentos caseros en trabajos de grupos bien coordinados, patrocinados por gobiernos y corporaciones, buscando colectivamente nuevos desarrollos y aplicaciones novedosas a técnicas ya conocidas.

1965: los primeros satélites

El producto de una interesante colaboración multinacional para el uso del espacio fue el lanzamiento y puesta en operación del primer satélite comercial de comunicaciones, el INTELSAT I, conocido también como el *pájaro madrugador*. El INTELSAT I tenía una capacidad de 240 circuitos telefónicos. Dos años después, se integraba un sistema global de comunicaciones vía satélite con la colocación en órbita de dos satélites adicionales de mayor capacidad, los INTELSAT II del pacífico y del atlántico, con lo cual se podía establecer comunicación telefónica (cerca de 720 circuitos para voz) entre cualquier ciudad del planeta. El INTELSAT V, puesto en órbita en 1980, podía procesar 12 000 llamadas telefónicas de manera simultánea, aparte de dos canales de televisión.

1988: la nueva tecnología en telecomunicaciones

El primer cable trasatlántico de fibras ópticas, el sistema TAT-8, fue puesto en operación entre estados unidos y gran bretaña. Sus propietarios: AT&T y un consorcio de 27 compañías y oficinas gubernamentales europeas. Podía transportar simultáneamente 40,000 conversaciones telefónicas, más de lo que podían transportar los otros cables y enlaces satelitales trasatlánticos combinados en ese entonces. Esto ocurrió 146 años después de que el primer conductor de señales subacuático fuera probado, en 1842 por S. Morse y E. Cornell, entre ambos lados del río Hudson; 137 años después de haber tendido cables que atravesaban el río Mississippi; 138 después del enlace Dover-Calais, y 122 después del primer cable exitoso que enlazaba Irlanda con New Foundland. Las comunicaciones internacionales vía satélite siguen creciendo con una tasa anual de 10%. El sistema INTELSAT cuenta con 16 satélites en operación; 11 de ellos pueden transmitir entre 12,000 y 15,000 canales de voz y adicionalmente, dos de televisión. De acuerdo con estas tendencias es posible suponer que dicha capacidad podrá ser expandida en el futuro a una cantidad cercana a los 100,000 circuitos telefónicos.

1996: la era de la computación y las redes de información

El crecimiento explosivo de redes que enlazan todo el planeta, computadoras que se comunican a velocidades de millones de bits por segundo, telefonía celular, localización global de personas, redes personales de comunicación, televisión de alta definición (HDTV), redes telefónicas interconectadas con redes de televisión por cable, realidad virtual, satélites de órbita baja, supercarreteras de información, etcétera.

Antecedentes

Los avances científicos logrados en las telecomunicaciones han requerido muchos años de experiencia, de innovación y de expansión: en los años cuarenta, se tenían objetivos de carácter militar y era de importancia secundaria el beneficio social que las comunicaciones tendrían como consecuencia; en los sesenta era muy satisfactorio poder conectar aparatos telefónicos en ambos extremos de un canal de satélite y sostener con inteligibilidad razonable una conversación telefónica; en los noventa se han incorporado los satélites a sistemas integrales de transmisión de información, con una gran variedad de medios de comunicación, tales como fibras ópticas y cables metálicos, permitiendo la satisfacción de una amplia gama de necesidades de comunicación entre las cuales están la transmisión de datos, las teleconferencias y, desde luego, las transmisiones de conversaciones habladas entre muchas otras. A mediados de los años noventa es imposible concebir muchas actividades humanas cotidianas sin el apoyo de las telecomunicaciones: fax, teléfono celular, televisión a colores, radiolocalización de personas, redes de computadoras, etcétera.

El desarrollo de los actuales sistemas de telecomunicaciones tiene su base en el invento de aparatos (telégrafo, teléfono, radio, televisión) de cuya evolución resultan los servicios que la tecnología de punta ofrece hoy en día: correo electrónico, televisión cromática, la transmisión de FM estereofónica, las transmisiones de televisión de alta resolución con sonido de alta fidelidad, el fax o las redes de computadoras.

El telégrafo

La telegrafía eléctrica ve sus inicios en el siglo XVIII (antes de que el estudio de la física y la electricidad alcanzaran la madurez suficiente para explotar sus posibilidades) como una alternativa mucho más rápida que cualquier medio conocido hasta entonces (antes de este siglo se utilizaba únicamente la telegrafía óptica: señales enviadas por medio de lámparas o antorchas). El telégrafo es un conjunto de sistemas que permiten enviar un impulso eléctrico a distancia, de modo que para enviar un mensaje, era necesaria una *codificación*, es decir: un método que permitiera transformar el mensaje en impulsos eléctricos que pudieran ser enviados, e interpretar los impulsos eléctricos recibidos para obtener el mensaje. Esta codificación se conoce con el nombre de *clave Morse* en honor a su inventor: Samuel Finley Breese Morse.

Esta clave definía dos *símbolos* o valores de un impulso eléctrico: punto (un impulso corto) y raya (un impulso largo). El código morse, asigna a cada letra del alfabeto, una determinada combinación de puntos y rayas única para ese símbolo (esto también posibilitó la concepción del telégrafo inalámbrico que usa ondas electromagnéticas para la transmisión). El telegrafista encargado de enviar el mensaje, utilizaba una llave para generar las combinaciones de impulsos correspondientes a cada letra del mensaje, una tras otra. En el punto en donde se recibía el mensaje, otro telegrafista estaba encargado de recibir los impulsos eléctricos en forma de sonidos e interpretarlos como letras, para construir el mensaje que se estaba transmitiendo. Al intervenir los telegrafistas, la información no podía ser totalmente confidencial para el remitente y el destinatario, además la generación de los impulsos era manual y muy lenta. Por otro lado, el telégrafo inalámbrico tenía un problema de confidencialidad, ya que el impulso transmitido viaja por la atmósfera, y cualquier persona que contara con el equipo de recepción adecuado, era capaz de recibir el mensaje, aún cuando no fuera el destinatario.

El teléfono

El teléfono es un aparato que convierte el sonido (vibraciones mecánicas del aire) en impulsos eléctricos que viajan a través de una red telefónica, y permite convertir esos impulsos eléctricos de nuevo en sonido. La diversificación en el uso del teléfono, se debe al hecho de que se puede aprovechar su misma red de transmisión para enviar impulsos que no necesariamente representen un sonido: pueden representar datos, una imagen, un video, etc. Su desarrollo fue posible gracias a los trabajos de muchos científicos norteamericanos y europeos. Michael Faraday, Charles Wheatstone y R. Willis, sientan las bases durante la primera mitad del siglo XIX. En 1876, Alexander Graham Bell y Elisha Gray (cada uno por su propia cuenta) realizando investigaciones sobre la transmisión de señales de voz por medio de cables, patentan casi simultáneamente una idea de lo que sería el teléfono, pero Bell ganó la aceptación de su patente (aún ante instituciones norteamericanas), a pesar de las protestas de Gray. En 1878, Thomas Alva Edison mejora la calidad de sonido en los prototipos de Bell, al incorporar transmisores de carbón. Hacia finales del siglo XIX, Bell funda su propia compañía (Bell Telephone Company), una de las más grandes y poderosas del mundo en la actualidad. Poco a poco, se ha ido desarrollando el sistema telefónico, introduciendo centrales de control, centrales retransmisoras, mejorando las técnicas de transmisión, utilizando cables de cobre reforzado en las comunicaciones, amplificadores, cables coaxiales cubiertos de polietileno, enlaces intercontinentales, multiplexión y superposición de señales, etc.

Radiodifusión

Dignos representantes de la evolución técnica del siglo XX, la radio y la televisión tuvieron sus inicios en las dos primeras décadas de este siglo. En este periodo fueron sentadas las bases para la radiotransmisión, que a la postre darían origen a las transmisiones comerciales de radio, así como a las de transmisión y recepción de señales de video, sobre las cuales se basa la televisión moderna. En este proceso participan principalmente E. H. Armstrong, con sus trabajos en radiorreceptores; V. Zworykin, quien trabajó en cámaras de televisión; J. L. Baird, quien por primera vez logró transmitir la imagen de un rostro humano a través de la televisión, con calidad *reconocible* (en blanco, negro y distintos tonos de grises). Las transmisiones regulares de estaciones de televisión también se iniciaron en esa época: en 1928 la WRNY de Nueva York; en 1929 la BBC (British Broadcasting Corporation) de Londres; la CBS y la NBC de Estados Unidos en 1931. En 1951 había en Estados Unidos más de 15 millones de televidentes. En 1941 se iniciaron transmisiones regulares de radio con la técnica FM (modulación de frecuencia), bajo la dirección de E. H. Armstrong.

Tanto para el sistema de radio como para el de televisión (conocidos genéricamente como sistemas de radiodifusión) es necesario que las señales originales, que contienen la información que ha de ser transmitida, sean convertidas en señales eléctricas, y a su vez en señales electromagnéticas, mismas que serán depositadas en la atmósfera para su transmisión.

Antecedentes

La Radio

La radio funciona de forma similar al teléfono: existe un dispositivo que transforma los sonidos en señales eléctricas, la diferencia es que estas son convertidas a su vez en señales electromagnéticas que son depositadas en la atmósfera. Cualquier persona que cuente con el equipo receptor adecuado, puede escuchar los sonidos transmitidos por una radiodifusora.

La Televisión

En el caso de la televisión, la señal que contiene la información es de mayor complejidad que la de la radio. Los tres elementos que contienen información acerca de las imágenes son los siguientes:

- La distribución de luminosidad, es decir, la forma en que aparecen luces (blanco), sombras (negros) y las distintas tonalidades de grises.
- La composición de la imagen en función de las tres dimensiones espaciales.
- Los movimientos de los elementos mencionados

Adicionalmente se tiene, desde luego, el sonido, mismo que recibe un tratamiento similar que en el caso de la radio.

A través de cámaras de televisión se integran los tres factores anteriores en una señal eléctrica equivalente, cuya amplitud varía con relación al tiempo. Esta conversión se realiza por medio de un proceso de barrido: la cámara genera un haz que se mueve horizontalmente de izquierda a derecha, detectando las variaciones en las características luminosas de las imágenes. Al llegar al extremo derecho de la imagen, regresa el haz a la izquierda, se mueve ligeramente hacia abajo, y repite el proceso hasta llegar a la parte inferior derecha de la imagen. En ese momento el haz regresa a la esquina superior izquierda de la imagen y repite el proceso. Esta información, interpretada como impulsos eléctricos, se transforma a su vez en ondas electromagnéticas para su transmisión. El número de líneas horizontales por imagen determina la calidad de la imagen reproducida, y existen diferentes normas internacionales al respecto.

La cantidad de información que se debe transmitir en un canal de televisión, requiere de un ancho de banda de 4.2MHz. La parte de audio necesita una banda adicional de 25kHz. Para evitar traslapes entre los canales (que puedan ocasionar interferencia por ejemplo entre los canales 4 y 5), se deja un espacio libre entre ellos, conocido como banda de guardia. Además, para garantizar que la imagen en el aparato receptor sea de buena calidad, que no se mueva aleatoriamente y que no aparezcan rayas horizontales o verticales en la pantalla se requiere de información adicional en la señal; esta información se conoce como de control o de sincronía; a través de ella se garantiza que el aparato receptor interprete cada imagen recibida como una imagen completa, es decir, que no tome y reproduzca la mitad de una imagen y la mitad de la siguiente para generar una imagen en el receptor.

La reproducción se hace invirtiendo las operaciones realizadas en la conversión inicial: se toma la señal eléctrica y se inyecta en un sistema (cinescopio) en el cual se realiza un barrido en la misma forma que la descrita anteriormente, generando a su paso puntos de diferente luminosidad e intensidad en la pantalla.

Hasta la década de 1930, los métodos de reproducción de imágenes eran mecánicos, tenían muy baja calidad de imagen, y podían reproducir únicamente 10 cuadros por segundo; ello motivó el uso de sistemas electrónicos para la transmisión de TV en blanco y negro a principios del S. XX. Hacia la segunda mitad del siglo, comienzan las transmisiones de TV a color (que mediante un diseño muy complejo, eran compatibles con sistemas de TV en blanco y negro)

Las investigaciones en torno a la televisión se aceleraron e hicieron posible el establecimiento de patrones universales de fabricación que evitarán las dificultades técnicas y económicas que se derivarían de una excesiva dispersión de modelos en el mercado. La independencia de los equipos estadounidenses y europeos cristalizó en la elección de dos sistemas de estándares diferentes que se extendieron por el resto del mundo: el estadounidense, el cual establece una relación de 30 cuadros de imagen por segundo y supone 525 líneas por imagen (estándar posteriormente homologado como NTSC: National Television Systems Committee) y el europeo (denominado PAL: Phase Alternation Fine), que adoptó 25 cuadros por segundo y 625 líneas por imagen.

Componentes de la radiodifusión: modulador, transmisor y antena

Una vez que se cuenta con las señales eléctricas equivalentes, las transmisiones tanto de radio como de televisión se realizan de una manera muy parecida. Se emplean sistemas de transmisión que consisten básicamente en las siguientes componentes:

Modulador

Su función consiste en trasladar el espectro de la señal a transmitir, a la banda (o frecuencia) en que debe realizarse la transmisión. Cada canal que se transmite, tanto en radio como en televisión, tiene una distinta frecuencia portadora, y esto es precisamente lo que ubica a un canal en el sitio adecuado del sintonizador del receptor. Por ejemplo, en radio (AM) la portadora de una señal que se recibe en 600kHz del cuadrante, tiene una frecuencia de 600kHz.

Transmisor

Su función consiste en amplificar la señal proveniente del modulador e inyectarla en el canal de transmisión (en este caso, mediante una antena).

Antena de transmisión

Se encarga de inyectar en la atmósfera la señal proveniente del transmisor.

Amplitud modulada (AM) y frecuencia modulada (FM)

Existen muchas formas de modular una señal encargada de transportar información (esta se conoce como portadora), y dependiendo del tipo de modulación, la información de interés será contenida en distintas partes de la señal modulada (la señal que contiene la información). Por ejemplo, en radio de amplitud modulada (AM) como su nombre lo señala, la modulación es de amplitud, y por tanto la información está contenida precisamente en la amplitud de la portadora modulada.

Antecedentes

En FM (modulación de frecuencia), la información está contenida en la frecuencia de la portadora; esto significa que a mayores amplitudes de la moduladora (que contiene la información original) se transmite una portadora modulada, cuya frecuencia es mayor que su frecuencia nominal, y cuando la moduladora tiene una amplitud menor la frecuencia de la señal transmitida también es menor.

Hacia las comunicaciones modernas

El pánico cimbró a los norteamericanos, parecía que la invasión a la tierra ya no era un mito, los marcianos habían llegado, y el año de 1938 jamás sería borrado de la memoria de los terrícolas... Pero no, no fue verdad, la sugestión que provocan los medios de comunicación quedaba demostrada, gracias a la transmisión radiofónica de una novela de H. G. Wells, dirigida por Orson Welles.

Para entonces el sueño de las telecomunicaciones dejaba de serlo, sólo quedaba esperar la evolución de los sistemas, dispositivos y técnicas para la transmisión de información a largas distancias de modo instantáneo. Esperar que la dinámica de lo cotidiano la envolviera.

A fines del siglo pasado era posible identificar unos cuantos acontecimientos científicos que a la larga seguramente generarían un importante cambio en el área. Hoy el número de acontecimientos, decisiones, inventos y desarrollos que contribuyen al avance de las telecomunicaciones es enorme y, de hecho, aumenta día con día. Parecería mentira, pero en un siglo (1850-1945) hubo menos cambios, innovaciones y avances tecnológicos que los que muy probablemente ocurrirán en la próxima década o posiblemente en el próximo año.

El resumen de los acontecimientos científicos que han permitido el desarrollo de las telecomunicaciones podría sonar pretencioso; sin embargo, mencionar todo lo que tecnológicamente se ha hecho para lograr el manejo de información que tenemos en la actualidad sería una labor ardua y casi imposible, así que la única intención es mencionar los hechos más importantes en la historia de los sistemas de comunicación.

1.4. El porqué de "a quien sea, a cualquier hora, en cualquier lugar, en cualquier dispositivo".

En la actualidad, viviendo en un mundo globalizado, en una era donde las telecomunicaciones digitales son de uso común, existe una competencia entre las empresas más directa que nunca. A lo largo de los últimos 50 años, la tecnología ha tenido un papel determinante en el éxito y expansión de prácticamente cualquier negocio: los sistemas de telecomunicaciones y de información son parte integral de la infraestructura de las empresas exitosas a nivel mundial. Hoy en día todo el mundo puede conocer sucesos que hayan pasado en cualquier punto del planeta en cuestión de segundos; con la misma rapidez se puede propagar información económica entre mercados bursátiles de diferentes países, o también intercambiar información estratégica entre 2 oficinas de alguna empresa que se encuentren en polos opuestos del planeta. Esta facilidad de difusión de información ha traído muchos cambios en las estructuras políticas, sociales y económicas de los países. Entre esos cambios, se encuentran la apertura comercial de las fronteras, lo cual facilita la entrada a un país de compañías transnacionales, y esto a su vez ha creado un ambiente de mucha mayor competencia en cualquier mercado del mundo. Cualquier empresa de cualquier nacionalidad que comercie en cualquier mercado, requiere contar con un gran nivel de competitividad para sobrevivir.

Sin embargo, desde hace algunas décadas, se ha incrementado la importancia y el uso de una herramienta que también ha determinado el éxito o el fracaso de muchos negocios en todo el mundo, esa herramienta se llama: información. El uso estratégico de la información es en la actualidad tan importante, como contar con recursos humanos calificados o una campaña publicitaria que impacte en el consumidor. La tecnología actual, nos brinda herramientas para hacer mucho más eficiente casi cualquier tipo de operación o transacción en los negocios, y proporciona mecanismos para intercambiar información estratégica entre oficinas, empleados o entre empresas que se encuentren inclusive en diferentes países. La cantidad y tipo de información que se puede compartir con este esquema, hace prácticamente imposible calcular, predecir o anticipar todas las circunstancias en las que un determinado sistema de información será demandado.

Existen diversos mecanismos por los cuales un sistema puede ser adaptado a esta creciente demanda, sin embargo últimamente ha surgido un problema en las empresas: por lo general, los departamentos encargados de los sistemas de información, se enfrentan a sistemas construidos en una forma un tanto obsoleta, monolítica, incompatible con otros sistemas y dependiente de ciertas tecnologías no estandarizadas, lo cual en la mayoría de los casos, amarra al sistema con una determinada marca de hardware o software, y esto ocasiona a su vez que el costo total de propiedad de la tecnología (TCO por sus siglas en inglés: Total Cost of Ownership) sea muy elevado. Por otro lado, la tecnología ha tenido un avance enorme en las últimas 5 décadas desde la aparición de la computadora, y ese ritmo de crecimiento parece ser cada vez más rápido, y aunque existen muchas empresas con la capacidad logística y económica de adecuar sus sistemas al mismo ritmo que la tecnología avanza, esto no es redituable, así sus intereses están centrados en comprar tecnología escalable, que no se vuelva tan rápidamente obsoleta, y pueda ser utilizada durante más tiempo para proteger así su inversión.

Ya que contar todo el tiempo con tecnología de punta no es rentable, finalmente los empresarios se ven en la necesidad de hacer más con menos.

Antecedentes

Hacer más significa satisfacer las demandas del negocio en cuanto a la información estratégica, y eso implica que debe estar *disponible para cualquier persona que la necesite*; puede ser información de algún proveedor que requiera un empleado de compras, o de algún cliente dirigida al área de ventas, o del inventario para alguna oficina de la empresa localizada en provincia o el extranjero; se pueden determinar costosas pérdidas para la empresa si esa información no está disponible para la persona que la requiere. Por ejemplo, si el empleado de compras no obtiene la información del proveedor a tiempo, puede haber una escasez de algún producto que un cliente solicite, y determinar adquirirlo con la competencia; el área de ventas podría buscar a un cliente para confirmar un pedido, o la oficina localizada en provincia podría requerir urgentemente a la oficina central algún producto que se le haya agotado y debe surtirse cuanto antes para entregar pedidos pendientes.

Además, esa información debe estar *disponible cuando se requiera*, por poner un ejemplo: una sucursal de un negocio está en Toluca y la matriz está en el D.F., pero trabajan con diferentes horarios (la oficina de Toluca cierra más tarde). Un cliente llega a las oficinas de Toluca y hace un pedido de un producto que está agotado. Es un horario en donde la oficina del D.F. ya está cerrada. La oficina de Toluca necesita saber si la matriz le puede mandar un pedido urgente del producto, para lo cual requiere conocer el inventario disponible de la casa matriz, aún cuando ya esté cerrada.

Lograr todo esto con menos, significa tomar en cuenta que las empresas cuentan con una determinada infraestructura en sus sistemas de información, y como se ha mencionado anteriormente, actualizar constantemente esa infraestructura no es redituable, de modo que se debe buscar una forma de adaptar o escalar esa tecnología a la demanda de nuestro sistema, así como buscar alternativas (ya sea de elementos de hardware o de software) que reduzcan el TCO, como por ejemplo: usar dispositivos únicamente de despliegue de información en lugar de dispositivos de procesamiento y despliegue de información, uso de terminales tontas en lugar de terminales inteligentes, uso de clientes delgados en lugar de estaciones de trabajo en red, ampliar el uso de sistemas distribuidos, etc. Finalmente, las empresas ven reducidas sus posibilidades de modernizar su infraestructura, y son obligadas a tener que hacer más con menos

Este escenario, es el que genera la necesidad de una plataforma y una arquitectura que permita el acceso a la información crítica las 24 horas del día, los 7 días de la semana, los 365 días del año, garantizando dicho acceso en beneficio de la empresa, pero no olvidemos que en un mercado globalizado y de enorme competencia, el valor estratégico de esta información en gran parte están determinados por *llegar a la persona, el lugar y el tiempo en que se necesite, obtenida por el dispositivo con que se disponga en ese instante*. De todo esto proviene la necesidad de que la información llegue *"a quien sea, a cualquier hora, en cualquier lugar, en cualquier dispositivo"* (*"anyone, anytime, anywhere on any device"*). En el mercado actualmente existen principalmente 2 arquitecturas integradas que ofrecen una solución a este problema: por un lado están Microsoft con su tecnología MS.NET, y Sun Microsystems con su arquitectura SunONE (basada en J2EE), mismas que se analizarán y discutirán en capítulos posteriores.

2.

Conceptos Básicos

2.1. Introducción

A partir del cero, los mayas lograron definir el ciclo lunar con un error inferior a 8 horas en 300 años. Los mayas no solo inventaron el concepto del cero, hecho que según los expertos precedió al descubrimiento equivalente realizado por los hindúes, sino que también desarrollaron un calendario a base de agudas observaciones científicas y cálculos muy precisos; éste calendario puede ser proyectado cronológicamente con validez por un lapso de 300 mil años. El desarrollo de la informática ha tenido una evolución tremenda. En sus comienzos, al parecer muy lenta, creando ruedas y palancas (Blaise Pascal en 1642), hasta el inicio del procesamiento de datos (Herman Hollerith en 1890), con la tabulación del Censo en los Estados Unidos. A partir de entonces, nos enfrentamos día a día con nuevas técnicas para el procesamiento de la información.

Pocas son las tecnologías que en tan breve lapso de tiempo hayan logrado tanto desarrollo y perfeccionamiento. Hacia 1951, la computadora ya se había introducido al mercado en forma comercial. Pero ni siquiera aquellos que mantenían un estrecho contacto con ella, estaban preparados para prever lo que vendría después. El desarrollo espectacular de la computadora (en términos de potencia y capacidad de procesamiento, así como de la variedad de aplicaciones en que se utiliza) constituyó una de las grandes sorpresas de nuestro tiempo. Algo que tal vez no sea sorprendente, es que el éxito alcanzado por la computadora es el resultado de la labor de muchas personas que han tratado de resolver múltiples problemas en diversos campos. O sea, que es la consecuencia natural del desempeño de nuestras actividades diarias.

En el transcurso de los siglos, la computadora ha tenido una evolución natural a partir de los primeros artefactos creados por el ser humano para contar, como el ábaco. En lugar de manipular las cuentas ensartadas en los hilos, hoy en día nos servimos de impulsos electrónicos para alcanzar los mismos fines. Y lo que es más importante: los costos de computación han descendido gracias a las innovaciones tecnológicas que han ocurrido en los últimos 30 años. Hoy es factible emplear computadoras en aplicaciones que hasta hace unos pocos años habrían sido poco o nada viables desde el punto de vista económico. Al igual que sucedió con el teléfono, la televisión, el automóvil y el avión, la computadora ha transformado el mundo. Y en forma similar a muchos de estos inventos, la computadora está construida en base a una tecnología que surgió con gran celeridad, especialmente después de la segunda guerra mundial.

Los componentes de la computadora han evolucionado en rápida sucesión: los contadores electromecánicos se transformaron en bulbos al vacío, el almacenamiento y la memoria magnética dieron paso a los circuitos microscópicos de estado sólido. En sentido estricto, la computadora se volvió dueña de su propio terreno cuando dos grupos de científicos, ingenieros y matemáticos (uno de ellos en la Universidad de Pennsylvania y el otro, de IBM) pusieron en práctica el concepto de que las instrucciones de la máquina podían almacenarse y procesarse igual que si se tratara de datos (esto se conoce también como máquina de Von Neumann). El desarrollo subsiguiente de la computadora provino de la aplicación de innumerables tecnologías nuevas y originales.

2.2. Historia de las computadoras

2.2.1. Historia de las Computadoras como Sistemas de Información

“No es admisible que los estudiosos y científicos en lugar de elaborar y confrontar nuevas técnicas, pierdan su tiempo como esclavos en las fatigas del cálculo, que podía ser confiado a cualquiera si se pudieran utilizar máquinas para ello”. G. W. Leibniz

En los últimos años, la computadora se ha trasladado de los confines de nuestra existencia hasta el centro de nuestra vida. Todos conocemos las ventajas y servicios que presta: su gran capacidad para almacenar y procesar vastas cantidades de información, para agregar datos a la información ya existente, para analizarla, recuperarla y transmitirla de un continente a otro a través de satélites o líneas telefónicas. Sabemos también que la computadora puede efectuar cálculos, establecer comparaciones, simular hechos y controlar operaciones científicas e industriales que se llevan a cabo en la realidad.

Igualmente sabemos que trabajan por medio de programas que indican las operaciones que el procesador central debe realizar de acuerdo con un código determinado; que poseen una memoria magnética que permite asentar todas las instrucciones y los datos con los que se le ha *alimentado*, transistores y circuitos integrados que se emplean para realizar las cuatro operaciones matemáticas básicas; sabemos que utiliza diferentes lenguajes o códigos para procesar esa información... Conocemos también algunos de los usos más comunes de las computadoras como la reservación de boletos de avión, el control de inventarios en fábricas o bodegas, el registro de historias clínicas de los pacientes de un hospital, la actualización de cuentas bancarias. En fin: difícilmente podríamos concebir el mundo moderno sin la computadora.

Pero todo este admirable progreso electrónico, toda esta complejidad matemática que se maneja casi naturalmente por medio de una computadora, es sólo el resultado de un proceso que esconde sus raíces en las primeras investigaciones de las culturas más antiguas de la humanidad. Desde los tiempos más remotos que registra la historia, las matemáticas han estado presentes. En cuanto el primer *homo sapiens* tuvo la necesidad de contar, enumerar y agrupar los diferentes elementos que constituían su mundo cotidiano, surgió la noción más elemental de las matemáticas.

En las primeras culturas que se desarrollaron sobre la tierra aparecen ya representaciones y sistemas numéricos, conceptos avanzados e instrumentos de contabilidad que, al paso de los siglos, vienen a constituir la base de las matemáticas y de las máquinas más avanzadas de nuestro tiempo. Caldeos, sumarios, babilonios y egipcios utilizaban todos los días números, cuentas, representaciones y procesos matemáticos en sus operaciones más elementales. Los mayas, los más brillantes científicos del pasado americano, inventaron el concepto del cero, principio de un sistema numérico tan complicado y perfecto que les permitió resolver complejísimo problemas matemáticos y astronómicos. Los árabes introdujeron el sistema decimal, el más usual de los sistemas numéricos que se utilizan hasta la fecha. Aportaron también el álgebra, punto de partida de la trigonometría, el cálculo integral y diferencial y otros tantos procesos matemáticos que constituyen las principales herramientas de los científicos modernos.

Además, nos legaron uno de los primeros instrumentos para contar de la historia: el ábaco. Los griegos también hicieron grandes descubrimientos y aportaciones en este campo. Simplemente citemos la concepción euclidiana del infinito, que permaneció vigente hasta nuestro siglo, en que Einstein la revolucionó con novedosas teorías que conforman las matemáticas modernas.

Muchas de estas nociones, que se remontan hasta los tiempos más lejanos de la historia de la humanidad, siguen siendo para nosotros los puntos de partida para efectuar los cálculos y las operaciones matemáticas necesarias para resolver los problemas de nuestra era. Y a la par de estos conceptos, como un complemento natural, fueron surgiendo las diferentes máquinas contables para llevarlos a la práctica. Ya hemos hablado del ábaco árabe e igualmente podríamos nombrar las tablas caldeas o el sistema de nudos inca. Pero conforme la humanidad ha ido progresando, ha efectuado nuevos descubrimientos científicos, geográficos y astronómicos con los cuales, obviamente, ha surgido la necesidad de producir máquinas cada vez más complejas que registren todas estas operaciones. Pero, ¿cuáles son estas máquinas? ¿Cómo han surgido? ¿Cómo funcionan? ¿Cuáles son los problemas específicos que han condicionado su aparición y su desarrollo?. El desarrollo social, artesanal y comercial que alcanzó Europa en el siglo XVII, condicionó la aparición de la primera máquina capaz de efectuar el cálculo automático. A continuación se presenta un resumen de desarrollo de la humanidad en esta área, comenzando en el siglo XVII

Siglo XVII: Las inquietudes

En el año 1642, a los 19 años, el matemático, filósofo y escritor francés Blaise Pascal inventa una máquina capaz de efectuar cálculos de manera puramente mecánica, y de realizar el acarreo. Posteriormente este tipo de acarreo automático constituye el principio fundamental de todos los instrumentos de cálculo, desde el odómetro del automóvil hasta las calculadoras de escritorio.

En 1671, el alemán Gottfried Wilhelm Leibniz proyecta una máquina que realiza automáticamente cálculos de tablas trigonométricas y astronómicas, sumas, restas, divisiones y multiplicaciones. Las máquinas de Pascal y Leibniz están limitadas a la comunidad científica del mundo, debido a su proceso de producción prácticamente artesanal. Su aplicación en problemas reales es aún lejana. Durante el siglo XVII y XVIII, las máquinas de cálculo mecánico comparten los mismos principios y sus mecanismos son perfeccionados continuamente.

Siglo XIX: Las bases de la Informática

En 1804, el francés Joseph-Marie Jacquard perfecciona la idea del mecánico Falcón, que un siglo antes había descubierto un nuevo sistema para automatizar algunas fases del trabajo de las máquinas tejedoras. El telar estaba guiado automáticamente en su movimiento por una serie de agujeros practicados sobre algunas tarjetas de cartón. Nace así la tarjeta perforada para transmitir a una máquina las instrucciones necesarias para su funcionamiento.

En el año 1820, el financiero francés Charles-Xavier Thomas de Colmar idea un dispositivo a base de piñones dentados que realiza multiplicaciones y divisiones basándose en el mismo principio de la calculadora de Leibniz. La realización de las operaciones hasta este momento, aún requiere intervención manual por parte de una persona.

Conceptos Básicos

En 1822, el matemático inglés Charles Babbage, concibe la idea de una máquina capaz de realizar cálculos automáticamente por medios mecánicos y recibiendo su información e instrucciones por medio de tarjetas perforadas. Esta máquina es la primera que incorpora conceptos parecidos a lo que son hoy los procesadores electrónicos.

Es hasta 1887 cuando el inventor francés León Bollé, perfecciona los mecanismos de multiplicación en una máquina, permitiendo hacerla directamente en lugar de hacerla en forma de sumas sucesivas.

En 1892, el suizo Otto Steiger fabrica la muy famosa y exitosa *millonaria* basado en los principios de la máquina de Bollé. Más de 4,500 máquinas de Steiger fueron vendidas durante 40 años entre 1894 y 1935

En 1899 el americano William S. Burroughs, hijo de un mecánico, inventa una máquina calculadora de multiplicación directa, que se vuelve muy popular en todo el mundo.

A principios de 1900 el enorme progreso de la técnica permite llevar a cabo, en gran escala, los proyectos que se habían acumulado en los siglos anteriores, formando un considerable patrimonio de ideas. Las máquinas calculadoras mecánicas, que posteriormente se transformaron en eléctricas, se producen en serie por importantes sociedades y se difunden rápidamente, sobre todo en el mundo de los negocios. Las calculadoras del primer período del 1900 presentan todavía posibilidades limitadas de aplicación y requieren, en mayor o menor medida, la continua intervención manual del hombre.

En el año 1887 aún no estaban terminados los cálculos del censo americano de 1880, elaborado manualmente por centenas de empleados. En previsión del censo para el año 1890, el experto en estadística Herman Hollerith, inspirado en la idea de los telares controlados por tarjetas perforadas de Joseph Marie Jacquard, inventa un sistema para representar los datos de una persona en tarjetas perforadas, que podían ser leídas y clasificadas eléctricamente. Los resultados del nuevo censo (en 1890) fueron obtenidos en solo dos años y medio, mientras que la población había crecido en un 26%. El éxito del censo americano provocó que las máquinas de Hollerith fueran empleadas inmediatamente para los censos austriacos y para efectuar el primer censo de la historia en Rusia, en el año 1896. Las máquinas de Hollerith fueron continuamente perfeccionadas y distribuidas en todo el mundo, y poco a poco se fue diversificando su uso, siendo utilizadas en el mundo de los negocios, en instancias gubernamentales, para información estadística, información de clientes, etc.

Siglo XX: El Desarrollo

Las máquinas fabricadas por la empresa fundada por Herman Hollerith (la Computing Tabulating Recording Company) existen en todo el mundo, y son producidas durante la primera mitad del siglo XX. Su éxito se debe a su perfeccionamiento para resolver grandes problemas contables y administrativos. En 1910, la empresa decide cambiar su nombre a International Business Machines (IBM).

Las nuevas tarjetas perforadas pueden contener información alfanumérica (y no únicamente numérica, como ocurría en un principio), las máquinas son más rápidas, capaces de procesar la información y presentar resultados impresos, ya que se introduce un mecanismo de memoria; ahora son incorporadas en compañías de ferrocarriles, teléfonos, petróleos, seguros, etc.

Hacia la década de 1950, estas máquinas eran de uso generalizado, inclusive en América Latina.

En 1936, el alemán Konrad Zuse construye un calculador electromecánico usando relevadores. Utiliza el concepto de representación binaria de los datos, y datos almacenados en cinta perforada.

En el año de 1944, el profesor Howard H. Aiken desarrolla en colaboración con IBM, la calculadora MARK 1 basada en relevadores y programa almacenado en cinta de papel perforada. Esta máquina representa la primera capaz de realizar las instrucciones contenidas en la cinta sin intervención humana.

En febrero de 1946, la Universidad de Pennsylvania pone en funcionamiento la computadora ENIAC (Electronic Numeric Integrator And Calculator): primera máquina completamente electrónica de la historia. Su construcción estuvo patrocinada por el ejército de E.U., y motivada por la posibilidad de realizar cálculos balísticos de su artillería a muy alta velocidad. Finalmente, la ENIAC facilitó el cálculo en una gran cantidad de trabajos científicos. Esta computadora no usaba ningún elemento mecánico para la representación de los datos, sustituyéndose por bulbos activados por impulsos eléctricos, ocupaba una superficie de 180 m², pesaba 30 toneladas, y usaba alrededor de 18,000 bulbos.

Desde 1943, el científico de origen húngaro John Von Neumann proyecta, en el Instituto de Estudios Avanzados en la Universidad de Princeton en los Estados Unidos, lo que hoy es universalmente reconocido como el verdadero prototipo de los modernos procesadores electrónicos. La nueva máquina, que se llamará EDVAC (Electronic Discrete Variable Automatic Computer) integra los conceptos de datos e instrucciones almacenados en memoria. Desde 1945 y hasta 1950 se proyectan diferentes prototipos tanto en los Estados Unidos como en Europa, basados en tecnologías de construcción de varios tipos y fueron identificados por las siglas: EDSAC, MADM, UNIVAC, SEAC, MANIAC, etc. La flexibilidad operativa introducida por el concepto de programa de Von Neumann permite el aprovechamiento de una máquina para las más diversas aplicaciones. Al inicio de los años 50's, estudios teóricos sobre procesadores electrónicos salen de los laboratorios universitarios, en donde se han originado toda una serie de prototipos aislados, que empiezan a interesar también a las industrias. De la fase puramente experimental se pasa a la producción de máquinas en diferentes modelos, destinadas a la venta comercial y así se inicia su difusión en las compañías y sociedades comerciales.

Un microprocesador es básicamente un elemento electrónico capaz de almacenar datos, representándolos en forma binaria, y operar en ellos por medio de un programa (serie de instrucciones almacenadas en la memoria que son interpretadas por la unidad de control del procesador), pudiendo o no, generar nuevos datos, también representados en forma binaria.

Conceptos Básicos

Para realizar las diferentes operaciones aritméticas y lógicas, el procesador electrónico se basa en algunas reglas muy simples, identificadas como *Álgebra de Boole*, nombre de un matemático inglés que las formuló cerca de la mitad del siglo XIX para sistemas de tipo binario. Los circuitos de cálculo de una máquina combinan entonces los varios impulsos eléctricos, referidos de tal manera que respetan las reglas binarias de la suma, de la multiplicación, etc.; usando el sistema binario, la máquina nunca calculará nada más complicado que 1+1. Cuando el procesador puede efectuar sustracciones, entonces es capaz de realizar la función lógica de la decisión.

El procesador también puede enviar y/o recibir datos de dispositivos externos (o *periféricos*), de forma que puede reaccionar ante cierto tipo de información. Así, puede saber cuando se presiona una tecla en el teclado, se mueve el ratón, puede leer una forma óptica, enviar a dibujar una imagen en el monitor, guardar datos en el disco duro o mandar imprimir un documento. Estos dispositivos periféricos, tienen la finalidad de facilitar la comunicación entre el procesador y los seres humanos, o inclusive con otros sistemas o procesadores.

Anteriormente, los datos y programas destinados al procesador, eran introducidos por medio de impulsos eléctricos comandados con una tarjeta o cinta perforada. Estos medios de almacenamiento (conocidos como almacenamiento secundario) han ido mejorando, se han compactado y se han hecho mucho más veloces, pasando por los tambores y cintas magnéticas, los discos flexibles, los discos duros, hasta los medios ópticos, como CD's y DVD's, e inclusive a los medios de almacenamiento secundario de estado sólido (circuitos integrados).

De la misma forma, han evolucionado las técnicas para programar los procesadores (es decir: definir la secuencia de instrucciones para que el procesador realice la tarea que deseamos que haga), siendo desde una representación binaria de instrucciones y datos en las tarjetas perforadas, hacia los lenguajes *ensamblador* (a base de mnemónicos para las instrucciones binarias), y ahora los lenguajes de alto nivel (Fortran, COBOL, C, C++, Basic, etc.).

Los procesos, son un concepto fundamental en los procesadores, pues es con un proceso como se identifica una determinada tarea que se está llevando a cabo. La forma en que se realizan los procesos, también se ha mejorado al paso del tiempo, para que, por ejemplo, un procesador realice un proceso en tiempo real (esto es, que genere los resultados de procesar una información tan rápido, que parezca que lo hace de forma instantánea), el multiproceso (dar la apariencia de que un procesador está realizando varios procesos al mismo tiempo), el multiusuario (permitir el uso de un procesador a más de una persona simultáneamente), etc. Todo ello gracias a las velocidades que han alcanzado los microprocesadores en la actualidad.

A la postre, este desarrollo ocasionó que los microprocesadores fueran los protagonistas de diversos logros de la humanidad, como la conquista del espacio, el poner a un ser humano en la superficie de la luna, la automatización de procesos industriales, el estudio de enfermedades y la creación de vacunas, el estudio del Genoma humano, etc.

Durante la segunda mitad del siglo XX, el desarrollo de los procesadores ha sido identificado por generaciones; la primera: los basados en bulbos; la segunda: los basados en transistores; la tercera: la basada en circuitos integrados; la cuarta: la alta integración de circuitos. Cada vez los procesadores son más rápidos, capaces de direccionar más memoria, y con aplicaciones también más diversas.

Siglo XXI: Nuevos procesadores.

La tendencia durante principios del siglo XXI, es buscar que un procesador haga más en menos tiempo, integrándole diversas tecnologías y arquitecturas. La alta integración de circuitos ha sido la técnica más importante para lograrlo, sin embargo, se ha hecho que una computadora sea una máquina modular, formada de varios sistemas funcionando sincronizadamente, y dando la posibilidad de armar una computadora tan potente como se desee (y al costo que se pueda pagar por ella), las velocidades del reloj en los procesadores hoy son del orden de los gigahertz, se incorporan memorias caché más rápidas e integradas dentro de los procesadores, la llegada de las redes de comunicaciones permiten compartir una máquina con más de un usuario, realizando cada uno más de una tarea simultáneamente, los sistemas operativos son cada vez más fáciles de usar, se crean nuevos lenguajes de programación más sencillos y más potentes, se idean mecanismos para que más de un procesador trabaje en una tarea al mismo tiempo (procesamiento paralelo), se programan ambientes de simulación, se crean clústeres de computadoras, etc. Todo esto, en su conjunto, implica en resumen: mejores computadoras, más potentes, más pequeñas, y más baratas.

Una Visión al Futuro

¿Qué nos reserva el futuro? Un número creciente de usuarios de computación comunicados a través de terminales unidas a redes de computadoras, mayores aplicaciones que facilitarán a la gente el desempeño de su trabajo en un mundo cada vez más globalizado. Pero hay mucha especulación con respecto a las innovaciones tecnológicas en el terreno de la computación. En realidad, la tecnología de semiconductores ha avanzado mucho en lo que se refiere a la dimensión cada vez menor de los módulos y a las técnicas de empaque de los mismos. De ahí la considerable disminución de costos y el aumento de la velocidad del tiempo de acceso. Consecuentemente, la relación precio/rendimiento de las computadoras ha aumentado diametralmente. Por otro lado, hubo un avance también significativo en el área de almacenamiento y rapidez de acceso a la información.

Esto gracias al aumento de la densidad de información registrada en espacios cada vez menores; como consecuencia, el costo de almacenar y disponer de los datos se redujo considerablemente. Lo importante es que el conjunto de evoluciones tecnológicas está causando una aceleración en la penetración de la informática en todas las actividades humanas; cada una de las técnicas arriba enunciadas tan sólo representa la culminación de un proceso claramente evolutivo. Lo que se espera es que ese proceso evolutivo siga acelerándose, aumentando la disponibilidad de información en espacios más diminutos y procesándolos a velocidades mucho mayores que las actuales, ya que el límite en la rapidez de acceso a los datos desde las computadoras es, teóricamente, el límite de velocidad electrónico, es decir: la velocidad de la luz.

Conceptos Básicos

En resumen, la tecnología es universal y lo que se espera en el futuro, es un proceso evolutivo acelerado y no una transformación que se convierta en una revolución tecnológica. En la era que pretendía ser de las naves espaciales, viajando a Júpiter, Saturno, etcétera, resultó ser la era de las comunicaciones y la computación. En los próximos 20 años, la tierra estará completamente computerizada; las transferencias electrónicas de información de país a país, ya no asombran pues se utilizan los medios más sofisticados de computación y comunicación.

Actualmente y en el futuro próximo, las computadoras de gran capacidad (no necesariamente de gran costo) estarán conectadas a minicomputadoras *inteligentes* y *no inteligentes*, y éstas a las terminales del usuario final; se puede anticipar que habrá más terminales y dispositivos personales que grandes supercomputadoras. Más que eso, la suma de las terminales podrá tener un valor de más del 70% del costo total del hardware en todo el mundo.

Ahora bien, el futuro muestra que la competencia en terminales será cada vez más amplia, permitiendo el uso de computadoras a centenares de empresas que hasta entonces no lo habían hecho debido a las altas inversiones requeridas previamente. Un reporte publicado por las naciones Unidas hace notar lo siguiente: "La tecnología juega un papel esencial en la reducción de las diferencias que existen entre los países desarrollados y aquéllos en desarrollo. Las computadoras ocupan un lugar importante en este contexto, puesto que muchas de sus aplicaciones tienen injerencia directa en algunos de los parámetros principales del proceso de desarrollo y reflejan algunos aspectos en donde la tecnología ha facilitado el crecimiento económico de países avanzados".

El desarrollo de la tecnología de la información a través de los semiconductores, almacenamiento magnético, circuitos integrados de silicio, sistemas de rayos láser, burbujas magnéticas, etcétera, son factores que cada día cambian radicalmente el proceso de la información, y es factible que en el futuro estos avances sean aún más espectaculares. Es por esto que tenemos un reto en América Latina, que consiste en saber aprovechar de una manera íntegra los recursos con los que actualmente contamos. La tecnología avanzada nos permite inventar nuevas formas para aplicar y realizar algunas tareas de vital importancia para el desarrollo y crecimiento de nuestros países. Conviene puntualizar, que la historia aún se sigue escribiendo, que el procesador electrónico es simplemente una máquina, pero para el hombre ha significado la liberación de los trabajos aburridos y repetitivos, y que le permite pensar y crear nuevas ideas para expandir su creatividad y su imaginación.

2.3. Internet y los sistemas de información

2.3.1. Internet hasta finales del Siglo XX.

Hace unos treinta años, la RAND Corporation, la primera fábrica de ideas de la América de la guerra fría, se enfrentó a un extraño problema estratégico. ¿Cómo se podrían comunicar con éxito las autoridades norteamericanas tras una guerra nuclear? La América post nuclear necesitaría una red de comando y control enlazada de ciudad a ciudad, estado a estado, base a base. Pero sin importar cómo esa red estuviera protegida, sus líneas y equipos siempre serían vulnerables al impacto de bombas atómicas. Un ataque nuclear reduciría cualquier red imaginable a pedazos. Entonces, ¿cómo sería controlada esa red? Cualquier autoridad central, cualquier núcleo de red centralizado sería un objetivo obvio e inmediato para un misil enemigo. El centro de la red sería el primer lugar a derribar.

La RAND le dio muchas vueltas a este difícil asunto, en secreto militar, y llegó a una solución atrevida. La propuesta de la RAND se hizo pública en 1964. En primer lugar, la red no tendría autoridad central, además sería diseñada desde el principio para operar incluso hecha pedazos. Los principios eran simples: se asumiría que una red era poco fiable en cualquier momento; todos los nodos en la red serían iguales entre sí, teniendo cada nodo autoridad para crear, pasar y recibir mensajes; los mensajes se dividirían en paquetes, cada paquete dirigido por separado; cada paquete saldría de un nodo fuente específico y terminaría en un nodo destino, recorriendo la red según unos principios particulares.

La ruta que tome cada paquete no tendría importancia, sólo contarían los resultados finales. Básicamente, el paquete sería lanzado de un nodo a otro, más o menos en dirección a su destino, hasta acabar en el lugar adecuado. Si grandes porciones de la red fueran destruidas eso simplemente no importaría; los paquetes permanecerían en la red en los nodos que hubieran sobrevivido. Este sistema de envío tan arbitrario podría parecer ineficiente en el sentido usual del término (especialmente comparado con el sistema telefónico, por ejemplo).

Durante los años 60, este proyecto marchó sin rumbo entre el RAND (*Research ANd Development*), el MIT (*Massachusetts Institute Of Technology*) y la UCLA (*University of California in Los Angeles*) de E.U. El Laboratorio Nacional de Física de Gran Bretaña preparó la primera red de prueba basada en estos principios en 1968. Poco después, la Agencia de Proyectos de Investigación Avanzada del Pentágono (ARPA) decidió financiar un proyecto de una red de (las entonces consideradas) supercomputadoras.

Entre 1969 y 1972, el número de nodos de esta red creció de sólo 1, a 37; para entonces, los administradores de las computadoras se percataron de que la red estaba siendo utilizada como medio de comunicación para fines personales (intercambio de archivos, mensajes, correos, etc.). Durante la década de los 70, la entonces denominada ARPANET creció rápida y fácilmente, gracias a su arquitectura descentralizada. El estándar de comunicaciones entonces era el *Network Control Protocol* (NCP), que posteriormente fuera sustituido por *TCP/IP*. El crecimiento (imposible de controlar) que experimentó ARPANET en las décadas de los 70's y 80's, terminaría siendo lo que conocemos como Internet, que se ha convertido en instrumento de información para todo mundo: desde instancias gubernamentales y militares, hasta su uso cotidiano en el hogar.

Conceptos Básicos

A finales de los 80's, también se comienza a hablar de los *hackers*, *crackers* y virus de computadoras. En 1990, se inventa el protocolo HTTP y se marca el inicio del World Wide Web, que facilita la navegación por la red; en 1993 se da a conocer el navegador Mosaic, pionero entre los navegadores Web. Aparecen los conceptos de HTML y URL's.

En 1994 se eliminan las restricciones comerciales sobre Internet, y a partir de 1995 hay un mayor crecimiento de la red de redes.

Es entonces que comienzan a aparecer más navegadores, sitios de servicios, de correo electrónico, de información, se propagan los virus y los gusanos. Se comienzan a crear una serie de conceptos que son ya toda una cultura alrededor de Internet.

En los últimos años del siglo XX y primeros del XXI, Internet se vuelve inalámbrico, pudiendo proporcionar servicios a teléfonos celulares, computadoras portátiles, y PDA's.

En 1998, se anuncia un proyecto conjunto de 34 universidades de EU.: Internet2, una iniciativa que permitirá crear una nueva red sobre nuevas arquitecturas de hardware y software para hacer de Internet un medio mucho más rápido, más seguro y más confiable para el transporte de información, basado en nuevas tecnologías y nuevos protocolos. Actualmente más de 160 universidades de todo el mundo trabajan en el proyecto Internet2.

2.3.2. Internet en el Siglo XXI

En la actualidad, y para decirlo en pocas palabras: Internet ha rebasado por mucho cualquier expectativa que se haya tenido sobre ella en los últimos 10 años. Su crecimiento, desarrollo, diversidad de aplicaciones, formas de interacción, etc. son muy numerosos y sería difícil mencionarlas.

Sin embargo, se pueden señalar algunas cosas interesantes que hoy en día se pueden hacer en la red de redes y que están llamando fuertemente la atención de las empresas por el potencial mercado que representan.

En el corto y mediano plazo, la red otorgará las facilidades para contar con tecnologías como la Automatización de Hogares, Dinero Virtual Electrónico, Bancos Virtuales, Educación Virtual y a distancia, Oficina Virtual, Realidad Virtual, Comercio Electrónico, Videoconferencias, Periódicos en línea, Servicios Web, etc.

Para todo eso, es necesario desarrollar los sistemas que garanticen la seguridad, confidencialidad y confiabilidad necesaria a los usuarios y brinde los recursos suficientes para enviar tal cantidad de información a sus destinos lo más rápidamente posible.

Todo esto es lo que se busca tener en las tecnologías que nos permitan implementar la red Internet 2.

2.3.3. Internet 2

La Red Internet2 (I2) espera ser 100 veces más rápida que la Internet actual y es desarrollada por un consorcio conformado por unas 300 universidades y más de 100 empresas, entre las que se encuentran Qwest Technologies, Cisco Systems, Nortel Networks, 3Com, AT&T, IBM, Microsoft, Sprint, MCI y muchos organismos estatales.

Internet2 está reservada para aplicaciones que requieren de una transmisión de datos casi instantánea; como ejemplo se pueden citar: la telemedicina, capacitación a distancia, manipulación simultánea de modelos computacionales, visualizaciones en tiempo real de cálculos efectuados en supercomputadoras remotas, vídeo bidireccional, teleconferencias, y procesamiento de imágenes satelitales, etc.

Esto significa que usuarios ubicados en diferentes lugares del mundo podrán ver las imágenes de vídeo en alta definición de un telescopio y observar todos simultáneamente el cosmos en tiempo real.

Pero seguramente esto sería poco si tenemos en cuenta que con esta tecnología se pueden crear laboratorios virtuales, desarrollar experimentos y compartir recursos científicos y médicos, como fotografías de células, tomografías, radiografías, etc., todo en cuestión de segundos. Internet 2 trabajará con el protocolo *IPv6* (versión mejorada del *IPv4*, que es el que se utiliza actualmente)

2.3.4. Internet Siempre Disponible

Para concluir sobre el futuro de Internet, se repetirá la frase, "Internet ha rebasado por mucho cualquier expectativa que se haya tenido sobre ella en los últimos 10 años", y a pesar de que en cierta medida ya hemos perdido la capacidad de asombro ante las novedosas posibilidades de la red de redes, su propia historia ha demostrado que habrá aplicaciones que imaginamos que habrá en 15 o 20 años, que veremos realizadas en 2 o 5 años.

El punto importante a señalar, es la tendencia a hacer disponible el Internet a cualquier persona en cualquier tipo de dispositivo, en cualquier lugar del mundo, las 24 horas del día, los 365 días de año, para ejecutar cualquier tipo de aplicación y tener disponible cualquier clase de información en el instante en que se requiera.

2.4. Nuevas Tecnologías

2.4.1. Dispositivos Portátiles (Celulares, PDA's, etc.)

Introducción

PDA (Personal Digital Assistant ó Asistente Digital Personal) denomina a cualquier dispositivo compacto que pueda caber en una mano o en un bolsillo, que proporcione procesamiento y capacidades de almacenamiento y recuperación de información, para uso personal o de negocios, generalmente se usa para tener en todo momento disponibles agendas, calendarios, direcciones, etc. Mucha gente usa el nombre de un par de productos populares como nombre del dispositivo genérico: *Palm*, derivado de las Palmtop de Hewlett-Packard y las Palm (o PalmPilot) de Palm Inc.

La mayoría de los PDA's incorporan un pequeño teclado y/o una pantalla sensible al tacto; algunos integran un reconocimiento de escritura manual, ya directamente en la pantalla o en un sensor. Los usos típicos de estos equipos incluyen la programación de actividades y horarios, así como almacenar direcciones y apuntes, aunque se han desarrollado múltiples aplicaciones adicionales que, si el hardware lo soporta, otorgan capacidades multimedia y de conectividad (correo electrónico, Internet, etc.). Muchos PDA's funcionan con un sistema operativo derivado de Windows: Windows Mobile, también de Microsoft. Otro sistema operativo altamente dominante es el Palm OS, de Palm Inc. Los demás incluyen sistemas operativos propios o derivados de alguno de éstos.

La década de los 60's y su visión futurista.

En los años 60's, Gene Roddenberry (el creador de *Star Trek*) ordenaba que no aparecieran papel o lápices en los sets de la nave espacial *Enterprise*. Los miembros de la tripulación utilizaban aparatos que probablemente son la inspiración de los dispositivos actuales. Comunicadores portátiles y *Tricorders* recopilaban y almacenaban datos y mensajes. En la década siguiente, Xerox basó parte de su trabajo sobre computadoras personales sobre una idea de Alan Kay: el Dynabook, una computadora personal interactiva portátil tan accesible como un libro o un cuaderno.

La década de los 80's: La concepción de los PDA's

En 1988, y tras el rotundo éxito de Mac II, Apple organiza un equipo de investigación para desarrollar una nueva generación de computadoras *más personales*. El dispositivo es bautizado con el nombre de *Newton*. El equipo tardó casi dos años en bosquejar lo que sería su propuesta: reconocimiento de escritura manual natural y funciones de comunicación.

La década de los 90's: De la PC al PDA.

Para 1991, en Apple los planes para el primer producto de Newton estaban listos: un pequeño dispositivo, que cabía en una mano, con funciones de planificación personal (calendario, libreta de direcciones, etc.) y de comunicación con el que se interactuaría mediante un lápiz.

El software *inteligente* para llevar a cabo esas funciones resultaba indispensable, pero aún así este producto fue presentado como una versión más compacta y económica del concepto original de Newton, que podía ser producida en serie. Su lanzamiento se retrasó hasta mediados de 1992.

En Enero de 1992, durante un discurso en el *Winter Consumer Electronics Show*, John Sculley, entonces la cabeza de Apple Computer, reclutado por el mismo Steve Jobs nueve años antes, acuñó la denominación de PDA al referirse a su visión de la computación personal próxima. Apple anuncia oficialmente el *Newton MessagePad*, que sería manufacturado por Sharp. A finales de año, Tandy y Casio anuncian el desarrollo de su PDA *Zoomer*, mientras corrían rumores de que Microsoft desarrollaba una contrapartida al Newton llamado *WinPad*. IBM presenta un tosco prototipo de un híbrido entre PDA y teléfono celular. Ese año también se funda Palm Computing Inc. Sus objetivos eran desarrollar software y hardware para las cada vez más versátiles agendas electrónicas, con miras a desarrollar su propio dispositivo portátil.

A principios de 1993 Motorola desarrollaba dos PDA's que usarían el sistema operativo del Newton, *Envoy* y *Gecko*. En Marzo, Amstrad lanza la primera PDA en el Reino Unido: se llamaba *PenPad* e incluía reconocimiento de escritura manual natural. En Agosto, Apple lanza finalmente el *Newton MessagePad*. Un mes después sale al mercado *ExpertPad*, de Sharp, un clon del Newton. En Octubre, Tandy y Casio lanzan *Zoomer*, y *PenPad* en los Estados Unidos. Terminando el año, IBM y Bellsouth anuncian *Simon*. Al año siguiente, Palm Computing comienza a desarrollar un dispositivo bautizado *Pilot*. Motorola y Sony anuncian oficialmente que sus PDA's *Envoy* y *Magic Link* contarán con el sistema operativo desarrollado por Apple. En Agosto, IBM y Bellsouth lanzan su *Simon* y en Septiembre es lanzado el *Magic Link* de Sony.

En 1995 es lanzado el *Envoy* y Panasonic exhibe un prototipo de PDA que usa el sistema operativo del *Newton*. Para este año, U.S. Robotics compra Palm Computing Inc.

En Marzo de 1996 Palm saca al mercado sus productos *Pilot 1000* y *Pilot 5000*. A diferencia del Newton y sus derivados, los sistemas de Palm incluían un reconocimiento de escritura manual no natural, teniendo el usuario que trazar cada letra una tras otra sobre un sensor.

Un año más tarde (1997) salen los modelos *PalmPilot Pro* (PalmPilot Professional) y *PalmPilot Personal Edition*, justo antes de que Palm se convirtiera en una subsidiaria de 3Com Corporation cuando ésta adquiere U.S. Robotics. En Septiembre, IBM lanza su *Workpad*, basado en Palm OS, sistema operativo que sería finalmente licenciado por Palm en Diciembre.

En Marzo de 1998 Apple Computer discontinúa la producción del Newton y se desentiende de él. Casi simultáneamente, Palm lanza su *Palm III*. Entre sus innovaciones destacaban su pantalla con iluminación posterior, comunicación infrarroja y un diseño exterior que caracterizaría estéticamente a los dispositivos de Palm desde entonces.

Conceptos Básicos

Durante 1999, TRG lanza *TRGpro*, el primer PDA expandible basado en Palm OS. Ese mismo año, Palm introduce su *Palm VII*, el primer PDA que incluía acceso inalámbrico a Internet, además de sus modelos *Palm IIIx*, *Palm V*, *Palm IIIe*, *Palm IIIe Special Edition* y *Palm Vx*.

Los PDA's del siglo XXI:

En el año 2000, Palm lanza su primer modelo con pantalla a color, el *Palm IIIc*, además de la versión expandible *Palm IIIxe* y sus siguientes modelos *Palm VIIx* y *Palm m100*. En Julio, 3Com y Palm se separan; las nuevas acciones de Palm se venden rápidamente. En septiembre, Motorola planeaba incorporar el Palm OS en una nueva línea de teléfonos celulares. Ese mismo año irrumpen otras compañías en el mercado, como Compaq y Hewlett-Packard.

Actualmente no se ven innovaciones a medida que los PDA's continúan desarrollándose. Las pantallas a color y capacidades multimedia son ya un estándar del mercado (Palm series m500 y Tungsten, Sony Clié, etc.). Los nuevos PDA's incorporan procesadores más poderosos y mayores capacidades de almacenamiento y de conectividad, son más livianos y funcionan por períodos de tiempo más prolongados. Palm, siguiendo la moda impuesta por los teléfonos celulares, lanza modelos con exteriores intercambiables. Esto no resulta del todo trivial, pues, actualmente, PDA's, handhelds y teléfonos móviles comparten muchas características; la tendencia apunta a combinar estos dispositivos en uno *definitivo*.

Software Utilizado

Estos sistemas comenzaron su desarrollo con los clásicos palm o PDA, que a pesar de ser novedosos y hasta cierto punto funcionales, nunca dejaron de ofrecer pocas prestaciones: no más que una agenda electrónica. Luego surgen nuevos dispositivos con mayores prestaciones y comienza paralelamente el desarrollo de los sistemas operativos y del software: hablamos de dispositivos de última generación como son los Handhelds (computadoras de mano), los PC's de bolsillo (Pocket PC's) que actualmente se ocupan o los Auto PC's que se perfilan como las computadoras que manejarán nuestros vehículos en un futuro.

Sistemas Operativos

Sin el software, una computadora no es más que un pedazo de metal sin utilidad. Con el software, una computadora puede almacenar, procesar y recuperar información, encontrar errores de ortografía en documentos e intervenir en muchas otras valiosas actividades. El software para computadoras puede clasificarse en general en dos clases: los programas de sistema, que controlan la operación de la computadora en si y los programas de aplicación, los cuales resuelven problemas para sus usuarios. El programa fundamental de todos los programas de sistema es el sistema operativo (SO), que controla todos los recursos de la computadora y proporciona la base sobre la cual pueden escribirse los programas de aplicación.

Desde su creación, las computadoras digitales han utilizado un sistema de codificación de sus instrucciones en sistema de numeración binaria, es decir con ceros y unos. Esto se debe a que los circuitos integrados funcionan con este principio: un transistor responde únicamente en función de que por él circule o no una corriente eléctrica.

En el origen de la historia de las computadoras, los sistemas operativos no existían y la introducción de un programa para ser ejecutado se convertía en un increíble esfuerzo que solo podía ser llevado a cabo por muy pocos expertos. Esto hacía que las computadoras fueran muy complicadas de usar y que se requiriera tener altos conocimientos técnicos para operarlas. Era tan complejo su manejo, que en algunos casos el resultado llegaba a ser desastroso. Además, el tiempo requerido para introducir un programa en aquellas grandes máquinas de lento proceso superaba por mucho el de ejecución y resultaba poco provechosa la utilización de computadoras para resolución de problemas prácticos. Se buscaron medios más elaborados para manipular la computadora, pero que a su vez simplificaran la labor del usuario. Es entonces cuando surge la idea de crear un medio para que el usuario pueda operar la computadora con un entorno, lenguaje y operación bien definido para hacer un verdadero uso y explotación de esta y sus recursos.

Un sistema operativo es el encargado de brindar al usuario una forma amigable y sencilla de operar, interpretar, codificar y emitir las ordenes al procesador central para que este realice las tareas necesarias y específicas para completar una orden. El sistema operativo, es el instrumento indispensable para hacer de la computadora un objeto útil. Bajo este nombre se agrupan todos aquellos programas que permiten a los usuarios la utilización de este enredo de cables y circuitos, que de otra manera serían difíciles de controlar. Un sistema operativo se define como *un conjunto de procedimientos manuales y automáticos, que permiten al usuario utilizar los recursos de la computadora*. Algunas de las funciones más importantes de los Sistemas Operativos son: ser la interfaz entre el usuario y el hardware de la computadora (Interpretar los comandos del usuario introducidos mediante dispositivos de entrada y presentar resultados mediante los dispositivos de salida), administrar y coordinar el funcionamiento del hardware y los dispositivos periféricos (impresoras, monitores, teclados, dispositivos de lectura y almacenamiento de datos como discos flexibles, discos duros, DVD-ROM's, etc.), controlar y recuperar la computadora de errores de software y hardware así como servir de entorno de desarrollo y ejecución de software de aplicación. Existen en la actualidad una muy numerosa diversidad de Sistemas Operativos para una (también numerosa) diversidad de procesadores y plataformas de cómputo.

Al igual que en las PC's, en los PDA's existe más de un Sistema Operativo. Es relevante entregar como aproximación una tabla comparativa del mercado de los 3 sistemas operativos para PDA's más importantes y dominantes del mercado, así como sus principales características.

Sistema Operativo	Mercado
Palm OS	41 %
Windows Mobile	39 %
Symbian	13 %
Otros	7 %

PALM OS

La compañía Palm, quien elaboró el sistema operativo Palm OS, fue fundada en 1992. En 1995 es adquirida por U.S. Robotics Corp., que a su vez es adquirida por 3Com en 1997, con lo que Palm pasa a formar parte de dicha compañía.

Conceptos Básicos

En septiembre de 1999, 3Com anuncia que Palm deja de formar parte de la corporación y pasará a ser totalmente independiente, hecho que se materializa el 2 de marzo del 2000. Siete meses después, Palm empieza a cotizar en el NASDAQ: la bolsa de empresas tecnológicas más importante del mundo. La compañía encabeza el mercado global de venta de PDA's, acaparando más del 75% del mercado mundial de venta de computadoras de mano. Buena parte del éxito hay que atribuírselo al sistema operativo de la compañía: Palm OS.

La plataforma está enfocada en posibilitar el acceso a cualquier tipo de información en cualquier lugar y momento. Además, el usuario puede organizar fácilmente sus citas y eventos, así como trabajar con aplicaciones y datos empresariales. Palm OS permite el acceso tanto a las páginas WAP como a las incluidas en la tradicional red de Internet, siempre que el PDA disponga de un navegador Web/WAP y un módem (o que la conexión se realice con la ayuda de un teléfono móvil). Una de las características esenciales de Palm OS es el gran volumen de aplicaciones que posee (más de 10,000), destinadas a cubrir todas las necesidades del usuario. Una cifra a todas luces superior a las más o menos 300 aplicaciones disponibles para Windows Mobile. Además, el software incluido en la Plataforma es compatible con archivos Excel, documentos de Word, e-book, e-mail y navegadores WAP y Web.

Aplicaciones en PALM OS.

En PalmOS existe un conjunto de aplicaciones estándar, existentes en prácticamente todo dispositivo que tenga instalado este sistema operativo: Procesador de textos, Calculadora, Escritura natural, Reconocedor de archivos, Agenda para organizar actos y citas, Control de gastos/optimizador de gastos, Directorio de contactos, Reloj mundial, Indicador de tareas por realizar, Correo electrónico, Buscador de archivos.

WINDOWS Mobile

Microsoft, el gigante de la informática dirigido por el multimillonario Bill Gates, ha visto un importante mercado en el segmento de PDA's, y ha creado un sistema operativo destinado a estos equipos. Su nombre es Windows Mobile y, tanto la interfaz gráfica como los programas que incluye, son muy similares al popular Windows. De hecho, prácticamente la totalidad de los programas incluidos en Windows Mobile son compatibles con los equipos de escritorio que trabajen bajo entorno Windows. De este modo, trabajar con un dispositivo portátil en lugar de una PC de escritorio es una actividad que casi no requiere aprendizaje adicional, debido a la similitud de ambas plataformas. Windows Mobile es una plataforma ideal para aquellos a los que les guste programar sus propias aplicaciones, ya que podrán utilizar sus conocimientos del Sistema Operativo Windows y los lenguaje de programación VisualBasic o Visual C++, pues las herramientas de desarrollo de Microsoft para Windows Mobile utilizan estos dos lenguajes y multitud de tecnologías comunes, como ActiveX, ADO y API's de Windows. Windows Mobile cuenta con una gran ventaja respecto a los otros dos sistemas operativos (Palm OS y Symbian OS): el monopolio y la experiencia que Microsoft posee en el mercado de los PC's. Pero también posee un aspecto desfavorable, al ser la última compañía en entrar en el negocio de los Sistemas Operativos para PDA's.

Una de las últimas acciones que ha llevado a cabo Microsoft para equipararse a sus competidores ha sido liberar el código fuente de Windows Mobile. De este modo, los desarrolladores podrán trabajar con el código y adaptarlo a las necesidades de sus empresas, aunque no podrán comercializar el producto resultante.

Aplicaciones en WINDOWS Mobile.

En Windows Mobile existen las aplicaciones básicas existentes en Windows: Internet Explorer, Microsoft Reader, Microsoft Word, Microsoft Excel, Juego solitario, Indicador de tareas por realizar, Reproductor multimedia, Calculadora, Notas, Buscador, Menú de acceso directo personalizable, Escritura natural, Reconocedor de archivos.

SYMBIAN OS (EPOC32).

El nacimiento del sistema operativo Symbian OS (EPOC32) se debe a la compañía fabricante de equipos de mano Psion. El sistema operativo fue creado en 1991. Trabajaba en equipos con CPU de 16 bits e incluía una aplicación para programar en OLP, un lenguaje orientado a objetos. En 1997 aparece EPOC32, que conserva las principales características de EPOC16, pero con arquitectura abierta (32 bits adaptables a cualquier tipo de CPU). Esto convirtió a EPOC32 en una plataforma compatible con equipos de otros fabricantes. En 1998 Psion Software se separa del Grupo Psion como Symbian OS, una alianza formada, en un principio, por Ericsson, Nokia, Motorola y Psion. En julio de 2001 el sistema operativo EPOC32 se bautiza como Symbian OS. Actualmente existen tres versiones de Symbian OS: Cristal, Pearl y Quartz. Cada una de ellas está incluida en los diferentes PDA's, según las características de los mismos. Estos pueden ser: con teclado y sin pantalla táctil, sin teclado y con pantalla táctil y con teclado y pantalla táctil.

Aplicaciones en SYMBIAN OS.

Aunque en Symbian OS no existe un conjunto de aplicaciones estándar, muy comúnmente podemos encontrar: Procesador de textos compatible con Word, Hoja de cálculo compatible con Excel, Correo electrónico POP3, SMTP, IMAP4 y MIME, Buscador de archivos, Agenda para organizar actos y citas, Calculadora, Directorio de contactos, Reloj mundial, Indicador de tareas por realizar, Java.

Mercado

El mercado de los sistemas operativos para PDA's esta en crecimiento, encontrándose inclusive versiones que provienen de Linux. Esto es una gran oportunidad para los usuarios, que contarán con alternativas que se acomoden de mejor manera a sus necesidades. Cabe mencionar que Linux ha dejado de ser el sistema operativo favorito únicamente de los Administradores de Sistemas y ha saltado a diferentes plataformas, como PC's, Macintosh, dispositivos móviles de captura de datos (lectores de códigos de barras y equipos de comunicación inalámbrica) y PDA's.

Software

Existe una extensa gama de software para los PDA's, desde manejo de la producción hasta juegos. Es precisamente esto lo que buscan los usuarios: tener en sus manos una herramienta poderosa y útil. Algunas de estas aplicaciones comunes a los PDA's son: contactos y calendarios, organizador de tareas, editor de archivos de MS Word y MS Excel, correo electrónico, acceso a Internet, capacidades multimedia, juegos, etc.

También podemos mencionar algunos accesorios de hardware para PDA's: teclados, fundas, estuches, audífonos, micrófonos, tarjetas de expansión, tarjetas de memoria, tarjetas de conectividad inalámbrica, conectividad a equipos de escritorio, etc.

PDA's como Herramientas de Información

Finalmente, hay que señalar que con la globalización social, cultural, económica y política del mundo, contar con información precisa en el lugar indicado y en el momento oportuno, puede determinar el destino definitivo de una empresa.

Los PDA's son dispositivos que por sus características de portabilidad, flexibilidad y facilidad de uso, además de su capacidad de conectarse por diversos medios a Internet, redes privadas, redes corporativas e intranets, son poderosas fuentes de información que se colocarán como parte muy importante en la infraestructura de negocios de cualquier empresa.

2.4.2. Sistemas de Comunicaciones Digitales

Desde los inicios de la comunicación hablada y escrita, el ser humano, por diversas causas, ha tenido la necesidad de transmitir información a grandes distancias. Ya sea para fines comerciales, bélicos, científicos o culturales, siempre se ha buscado que esa información viaje largas distancias en el menor tiempo posible. La historia del nacimiento de las comunicaciones ya ha sido discutida, pero ahora se hará una breve introducción a las comunicaciones digitales, qué son, cómo funcionan y en dónde se aplican.

Conversión analógico-digital de una señal

El proceso de una conversión analógico a digital (también conocido como proceso de digitalización), implica convertir una señal continua en tiempo y amplitud, a una señal discontinua (o discreta) en tiempo y amplitud. Una señal analógica puede verse en una gráfica como una línea continua con variaciones de amplitud de precisión infinita. Una señal digital, se puede ver como una sucesión de puntos equidistantes en tiempo, cada uno con un valor de amplitud que está contenido en un conjunto finito de valores posibles. Esto implica un proceso de cuantificación (mapeo de la infinidad de valores analógicos, al conjunto finito de posibles valores digitales)

Codificación de un mensaje

La codificación del mensaje se refiere a la transformación que se le debe aplicar, de forma que pueda ser transmitido por el canal de transmisión disponible.

Criptografía de un mensaje

La criptografía de un mensaje se refiere a un conjunto de reglas, métodos y algoritmos previamente definidos que permitan *cifrarlo*, de modo que únicamente el receptor que conozca dichas reglas y cuente con los medios adecuados, será capaz de *descifrar* la información para interpretar correctamente el mensaje. Cualquier otro receptor que no conozca las reglas de criptografía del mensaje o no cuente con el medio adecuado para descifrarlo, será incapaz de descifrarlo e interpretarlo correctamente. Idealmente, los medios para descifrar la información son conocidos únicamente por el destinatario de la misma.

Sincronización entre transmisor y receptor

La sincronización entre transmisor y receptor, significa definir reglas específicas para que la velocidad a la que el mensaje es enviado, sea la misma a la que es recibido, previniendo así pérdidas (por que el receptor sea más lento que el transmisor) o duplicidad de datos (porque el receptor sea más rápido que el transmisor) en el contenido del mensaje.

Necesidad de un protocolo

Un protocolo define un conjunto de reglas necesarias para permitir la comunicación. Esto es: al entablar una comunicación, el protocolo nos especifica cuales son los pasos que se deben seguir, y en qué orden, tanto por el transmisor como por el receptor para lograr entablar una conexión y el diálogo correspondiente a la transmisión de la información.

Conceptos Básicos

Presencia de distorsión y de ruido en las comunicaciones

La distorsión y el ruido en un canal de comunicaciones, son diversos factores que ocasionan que el mensaje que llega al destinatario sea en cierto modo y en cierto grado, diferente al mensaje original transmitido por el receptor, lo cual puede ocasionar que algún dato, alguna parte del mensaje, o inclusive el mensaje completo, sean interpretados de forma incorrecta.

Detección y toma de decisiones en las comunicaciones

En todo sistema de comunicaciones, el receptor puede verse en el caso de tener que decidir el valor de algún dato que le parezca ambiguo. Dependiendo del sistema específico de que se trate, el proceso de decisión implicará cierto grado de incertidumbre (es posible que el receptor asigne un valor al dato que no coincida con su valor en el mensaje original que le fue transmitido).

Modificaciones para mejorar las comunicaciones

Confirmando recepción correcta y/o solicitando una retransmisión

Cuando el receptor recibe un mensaje con un dato cuyo valor es ambiguo, y éste tiene que decidir su valor, se dice que toma un papel de receptor pasivo. Por el contrario, si el receptor tiene la posibilidad de confirmar la recepción correcta, o solicitar una retransmisión del mensaje en caso de encontrar ambigüedades, se dice que toma un papel de transmisor activo.

Introducción de repetidoras

En diversos sistemas de comunicaciones, la distancia que puede cubrir un mensaje puede estar determinada por las limitaciones propias del canal de transmisión (dispersión de un haz de luz, atenuación de una señal eléctrica por un conductor, etc.). En estos casos se introduce el uso de repetidoras: dispositivos cuya función es recibir el mensaje desde el transmisor, analizarlo, amplificarlo y volverlo a enviar en dirección de su destinatario. De esta forma, el proceso de transmisión se puede dividir en varias etapas, y se puede lograr transmitirlo a una mayor distancia.

Agregando redundancia

Cuando un mensaje contiene codificados más datos que los estrictamente necesarios para transmitir la información, se dice que contiene información redundante. Esta información redundante puede hacer que la cantidad de datos a ser transmitidos sea mayor, pero también puede ser usada como mecanismo de detección de errores de transmisión y/o recepción por el destinatario.

Transmisiones punto a multipunto

En muchos sistemas puede desearse que un mensaje transmitido, en lugar de ser recibido por un solo destinatario en una transmisión punto a punto (como puede ser el telégrafo, por ejemplo) sea recibido por varios destinatarios (como es el caso del radio o la televisión) en lo que se denomina transmisión punto a multipunto.

Esto implica que cada destinatario debe contar con un equipo receptor capaz de obtener el mensaje desde el medio de transmisión, además de que éste receptor debe conocer los protocolos, reglas y medios adecuados para poder decodificar y en su caso, descifrar la información.

Sistemas de comunicaciones con transformaciones

En 1949, C. E. Shannon propuso lo que llamó una “*teoría matemática de la comunicación*”, donde analiza las siguientes cuestiones fundamentales: ¿Cómo se puede medir la cantidad de información contenida en un mensaje?; ¿Cómo se puede medir la capacidad que tiene un canal para transmitir información?; ¿Cuáles son las características deseables para un codificador?, y cuándo este proceso se realiza en forma eficiente ¿cuánta información puede ser enviada a través de un canal?; ¿Cuáles son las características generales de los procesos de ruido y cómo afectan la calidad de los mensajes recibidos en el receptor?

Los conceptos y las ideas contenidos en dicha teoría han servido desde su publicación como semillas para la mayoría de los trabajos modernos de las comunicaciones digitales. Se postulan definiciones de índices de desempeño, y se analizan diversos mecanismos de procesamiento de información: un buen número de los resultados actuales giran alrededor de la obtención, el diseño y la realización electrónica de sistemas y dispositivos electrónicos que alcancen (o por lo menos se aproximen) tanto como sea posible a los desempeños predichos por Shannon. Para explicar los conceptos establecidos por Shannon partimos de su sistema conceptual, cuyas componentes se explican con el siguiente diagrama:

Fuente > Codificador de la Fuente > Codificador del Canal > Canal de transmisión >>>...>>> Canal de transmisión > Decodificador del Canal > Decodificador destino > Destino

Considerando sistemas digitales, en el bloque que sigue a la fuente (es decir, el codificador de la fuente), se realiza la función de convertir el mensaje original (el cual no necesariamente es de tipo digital o binario) en un mensaje binario, esto es: en una sucesión de *unos* y *ceros*. A su salida se tiene conectado el codificador del canal. Su función es proteger la información transmitida contra los efectos y fenómenos a que está expuesta al viajar a través del canal. Esto se logra agregando redundancia a la información transmitida, con el objeto de que en el lado del receptor se pueda identificar y en su caso corregir esta situación.

Redes de telecomunicaciones

Un Sistema de Telecomunicaciones consiste en una infraestructura física a través de la cual se transporta la información desde la fuente hasta el destino, y con base en esa infraestructura se ofrecen a los usuarios los diversos servicios de telecomunicaciones. En lo sucesivo se denominará *red de telecomunicaciones* a la infraestructura encargada del transporte de la información. Para recibir un servicio de telecomunicaciones, un usuario utiliza un equipo terminal a través del cual obtiene entrada a la red por medio de un canal de acceso.

Cada servicio de telecomunicaciones tiene distintas características y puede utilizar diferentes redes de transporte, por lo tanto para acceder a todos ellos, el usuario puede requerir de distintos equipos terminales.

Conceptos Básicos

La principal razón por la cual se han desarrollado las redes de telecomunicaciones es que el costo de establecer un enlace fijo entre dos usuarios de una red sería muy elevado, sobre todo considerando que no todo el tiempo todos los usuarios se comunican entre sí. Es mucho mejor contar con una conexión dedicada para que cada usuario tenga acceso a la red a través de su equipo terminal, pero una vez dentro de la red los mensajes utilizan enlaces que son compartidos con otras comunicaciones de otros usuarios.

En general se puede afirmar que una red de telecomunicaciones consiste en los siguientes componentes:

1. Un conjunto de nodos en los cuales se procesa la información.
2. Un conjunto de enlaces o canales que conectan los nodos entre sí y a través de los cuales se envía la información desde y hacia los nodos.

Desde el punto de vista de su arquitectura y de la manera en que transportan la información, las redes de telecomunicaciones pueden ser clasificadas en Redes Conmutadas y Redes de Difusión.

Redes conmutadas

La red consiste en una sucesión alternante de nodos y canales de comunicación, es decir: la información es transmitida a través de un canal, llega a un nodo, el cual a su vez la procesa para enviarla por el siguiente canal que llega al siguiente nodo, y así sucesivamente. Existen dos tipos de conmutación: conmutación de paquetes y conmutación de circuitos. En la primera, el mensaje a transmitir se divide en varias partes que se encapsulan en un paquete con información suficiente sobre su origen, su destino y la forma en que se debe volver a ensamblar. Estos paquetes se envían en una red, viajando de nodo en nodo, siguiendo cada paquete un camino probablemente diferente. En el lado del receptor, se reciben los paquetes posiblemente de diferentes fuentes y no necesariamente en el orden en que fueron enviados, tras lo cual el mensaje original es reensamblado. En la conmutación de circuitos, intenta mantenerse siempre un mismo canal y una misma ruta entre emisor y receptor (como ejemplo tenemos el sistema telefónico)

Redes de difusión

En este tipo de redes, se concentran varios transmisores que envían todos sus mensajes en un mismo medio de transmisión, y los receptores se encargan de sacar de este medio únicamente la información que les interesa a través de su propia terminal. Existen redes como el radio o la TV, en donde los receptores son pasivos (sólo reciben información, pero no pueden transmitirla); por otro lado, tenemos el teléfono como el mejor ejemplo de un receptor activo (un usuario puede recibir y enviar mensajes desde la misma terminal).

Una red es pública, si cualquier usuario tiene acceso a ella, o privada, cuando su uso se restringe a los propietarios de la misma, y se niega a terceros. Una característica importante de una red, es su área de cobertura geográfica (el área en donde los usuarios pueden conectarse y usar sus servicios), y puede ser desde una red de cobertura local en una oficina o un edificio (Local Area Network, LAN), hasta una red de cobertura entre ciudades, estados, ó países.

Canales

El canal es el medio físico a través del cual viaja la información de un punto a otro. Sus características son importantes para una comunicación efectiva, ya que de ello depende en gran medida la calidad de las señales recibidas en todos los nodos de un circuito de comunicación. Los canales pueden pertenecer a una de dos clases:

1) Canales que guían las señales que contienen información desde la fuente hasta el destino, por ejemplo: cables de cobre (a 4Mbps en par trenzado), cables coaxiales (hasta 500Mbps) y fibras ópticas (a 10Gbps).

Los cables de cobre son el medio de transmisión más utilizado. Es un medio que produce atenuación en las señales eléctricas que conduce, por lo que generalmente requiere de repetidoras a distancias más o menos cortas. Los cables se recubren con una protección o blindaje que los protege de ruido electromagnético externo. En los últimos años también ha aumentado el uso de fibra óptica como canal de transmisión de señales ópticas: la fibra óptica es un material mucho más ligero, no produce atenuación de la señal (aunque sí permite su dispersión), de modo que la distancia entre repetidoras puede ser mucho mayor, y aunque son susceptibles a ruido luminoso, son inmunes a ruido electromagnético, además de que permiten transmisiones mucho más rápidas que los canales de cobre.

2) Otro tipo de canales difunden la señal sin una guía (utilizan el aire como canal de transmisión), a los cuales pertenecen los canales de radio, canales de microondas y enlaces satelitales. Las microondas utilizan antenas de transmisión y recepción de tipo parabólico para transmitir con haces estrechos y tener mayor concentración de energía radiada. Los satélites funcionan de forma similar a las microondas, salvo que una de las antenas emisora/receptora se encuentra en un satélite generalmente geoestacionario, lo cual es una ventaja por área de cobertura, pero una desventaja por el tiempo que tarda en viajar la señal.

Para funcionar, estas antenas se basan en un principio geométrico: la geometría de una parábola es tal, que una emisión que llega a la parábola paralela a su eje, es reflejada y pasa por su foco. De la misma forma, una emisión que es emitida en su foco, al incidir sobre la superficie parabólica, es reflejada paralelamente a su eje. Una antena parabólica recibe su señal paralela al eje, y se refleja a un receptor localizado en su foco geométrico, donde también está localizado el emisor que genera la señal que viaja también paralela al eje de la antena. Estas señales se denominan unidireccionales. Una señal de una transmisora de radio viaja en todas direcciones, y por ello se denomina *omnidireccional*.

Algunas de las aplicaciones (clasificadas por rangos de frecuencia) de las ondas de radio se mencionan a continuación:

- 30-300kHz, LF (*low frequency*): Baja frecuencia. Navegación aérea y marítima.
- 300-3000kHz, MF (*medium frequency*): Frecuencia media. Navegación, radio comercial AM, enlaces privados fijos y móviles.
- 3-30MHz, HF (*high frequency*): Alta frecuencia. Radiodifusión onda corta enlaces fijos y móviles.

Conceptos Básicos

- 30-300MHz, VHF (*very high frequency*): Muy alta frecuencia. Televisión y radio FM, enlaces fijos y móviles.
- 300-3000MHz, UHF (*ultra high frequency*): Frecuencia ultra alta. Televisión y microondas, navegación, meteorología.
- 30GHz, SHF (*super high frequency*): Frecuencia super alta. Microondas y satélite, radionavegación.
- 30-300GHz, EHF (*extra high frequency*): Frecuencia extra alta experimental

Esta clasificación es muy generalizada, y existen otras muchas aplicaciones. La mayoría de los sistemas de telecomunicaciones digitales combinan diferentes tipos de canales en una transmisión (en tramos se usan microondas, en otros señales por cables, en otros enlaces satelitales, etc.).

Nodos

Los nodos son parte fundamental en cualquier red de telecomunicaciones, son los encargados de desempeñar las diversas funciones que permiten realizar el enlace, la transmisión y recepción de los datos. La mayoría de los nodos de casi toda red de telecomunicaciones, realizan las siguientes funciones:

- a) Establecimiento y verificación de un protocolo
- b) Transmisión y uso eficiente del canal.
- c) Transformación del mensaje de forma que pueda ser *vaciado* dentro del canal de transmisión.
- d) Recuperación de algún error de transmisión.
- e) Formateo de la información para su tránsito por diferentes redes con diferentes protocolos.
- f) Enrutamiento para garantizar que los paquetes lleguen a su destino correcta y los mas rápidamente posible.
- g) Repetición de la transmisión en caso de que el receptor detecte problemas ó inconsistencias en el mensaje recibido.
- h) Direccionamiento para identificar la dirección de destino de un paquete.
- i) Control de flujo para optimizar el uso del canal y evitar su saturación.

Las funciones que se han descrito, son las más importantes que deben tener instrumentadas los nodos de una red más o menos compleja. Por ejemplo, si una red consiste solamente en dos nodos, es evidente que no se requieren funciones tales como direccionamiento o enrutamiento en cada uno de ellos.

El valor de las telecomunicaciones es el conjunto de servicios que se ofrecen por medio de las redes y que se ponen a disposición de los usuarios.

Servicios modernos de telecomunicaciones

A continuación, se compararán algunas características de ciertos sistemas de telecomunicaciones. Existen muchas formas diferentes de clasificar estos sistemas, pero para nuestro caso en particular se utilizan los siguientes criterios de comparación:

- a) Tipo de red. Se hablará del tipo de medio de transmisión que utilizan redes públicas.
- b) Cobertura. Se refiere al área geográfica en donde los usuarios pueden utilizar los servicios de la red. La cobertura puede ser caracterizada como local, regional o nacional.
- c) Interconexión. La capacidad de una red de conectarse a otras redes diferentes, y poder ofrecer más servicios.
- d) Direccionalidad. Se refiere al papel del receptor: pasivo (transmisión unidireccional: sólo recibe información [U]) ó activo (tiene capacidad de recibir y enviar información en una transmisión bidireccional [B]).
- e) Punto-multipunto. Nos dice la forma en que es hecha la difusión: si entre un emisor y un receptor (punto a punto, P-P), o entre un emisor y varios receptores (punto a multipunto, P-MP).
- f) Tipo de información. La información enviada por el canal, puede ser analógica (A), Digital (D) o cabe la posibilidad de que el mismo canal permita transmisiones de ambos tipos (A/D).
- g) Privacidad. Puede verse como el nivel de confianza que un usuario emisor tiene, de que la información que está transmitiendo no podrá ser recibida e interpretada más que por el receptor al que está destinada. Hemos asignado 3 niveles de confiabilidad en la privacidad de los servicios que estamos comparando (1 = baja, 2 = media, 3 = alta).

Evidentemente existen elementos adicionales que podrían ser incluidos (por ejemplo, costo del servicio, costo de los equipos terminales, características del canal de acceso), pero la lista anterior es suficiente para ilustrar diferencias entre los servicios presentados en el siguiente cuadro.

Conceptos Básicos

Servicios de Telecomunicaciones							
	Red			Dir.	Pto.	Info.	Priv.
Servicio	Tipo	Cobertura	Interconexión	U/B	P-P / P-MP	A/D	1,2,3
Telefonía	Cable	Nacional	Cobertura global; con otras redes nacionales	B	P-P	A/D	1,2,3
Telefonía móvil	Radio	Local	Con red telefónica	B	P-P	A	1
Telefonía celular	Radio	Regional	Con otras redes celulares y telefónicas	B	P-P	A/D	1
PCN/PCS*	Radio	Local	Igual que celular	B	P-P	D	3
Datos	Cable Fibra Radio Satélite	Regional	Sí	B	P-P / P-MP	D	1
TV difusión	Radio	Regional		U	P-MP	A/D	1
TV cable	Cable	Regional	Sí	U/B	P-P / P-MP	A/D	2
TV restringida	Cable o Radio	Regional	Sí	U	P-MP	A/D	2
Localización de personas	Radio	Regional o nacional	Otras redes de radiolocalización	U	P-MP	D	2
Comunicación Radio especializada de flotillas	Radio	Regional	Otras redes y red telefónica	B	P-MP	A/D	1

* PCN = Personal Communication Networks; y PCS = Personal Communication Systems.

2.4.3. Necesidades y tendencias en la actualidad: Sistemas de Información Inalámbricos

Radiodifusión de señales: de la radiotelefonía a la telefonía celular.

Dentro de los servicios de radiodifusión, existen dos clases: los bidireccionales y los unidireccionales. En el conjunto de los primeros se describirá la operación de la telefonía celular y su evolución hacia los servicios personales de comunicación (denominados PCN por su nombre en inglés: Personal Communication Networks o PCS: Personal Communication Systems). Para el caso de señales unidireccionales, podemos mencionar la radiolocalización de personas, pero no se discutirá por estar fuera del alcance del presente documento.

La radiotelefonía celular surgió como un avance importante de la radiotelefonía tradicional. En esta última, los conceptos son muy similares a los de la red telefónica pública, con la excepción de que el acceso por parte del usuario es por medio de un canal de radio, con sus equipos terminales correspondientes.

La radiotelefonía tradicional (donde las antenas de las centrales eran enormes, consumían demasiada potencia, y el uso del espectro de ondas electromagnéticas no era eficiente) fue evolucionando hacia el concepto de *telefonía celular*, con base en dos objetivos: aumentar la calidad de los servicios ofrecidos, y aumentar (compartiendo las frecuencias) la utilización del espectro radioeléctrico, lo cual dio como resultado el aumento del número de posibles usuarios de una sola red.

Las ventajas que se esperaba que la telefonía celular tuviera sobre la red telefónica tradicional son:

- a) Los equipos terminales (es decir, los aparatos telefónicos) son portátiles y no requieren de un enlace de cable para tener acceso a la red telefónica.
- b) Un equipo terminal puede desplazarse dentro del área de cobertura sin interrumpir la comunicación.
- c) Por medio de un equipo de telefonía celular se pueden establecer conversaciones con equipos telefónicos conectados a la red telefónica tradicional.
- d) El número de usuarios de una red puede aumentar casi sin límite debido a la posibilidad de reutilizar frecuencias, de reducir tamaños de células y de explotar adecuadamente las complejas técnicas de codificación.

Imágenes para el futuro

Las tecnologías presentes y futuras relacionadas con las telecomunicaciones nos inducen a pensar de una manera diferente a como lo hemos hecho en el pasado. Los avances en tecnologías digitales y en transmisiones por fibras ópticas permiten hablar ahora de velocidades de transmisión y de conmutación de menos de una mil millonésima de segundo.

Conceptos Básicos

Podemos entonces identificar las siguientes tendencias en los sistemas y los servicios de telecomunicaciones:

- Mayor conectividad entre redes, y por consecuencia, entre usuarios y servicios.
- Aumento de las comunicaciones móviles.
- Calidad digital de los servicios ofrecidos.
- Comunicación personal para cada habitante del planeta.

Es indispensable que los servicios sean accesibles a todos los usuarios, para lo cual será necesario que todos los servicios y terminales, incluyendo la combinación de más de un servicio, sean muy amigables y accesibles para la mayoría de la población. Las velocidades de las redes irán creciendo junto con la calidad de los servicios, y al ser accesibles a más personas, su costo será más bajo.

Por sus características de capacidad, con seguridad serán las fibras ópticas los medios predominantes en aquellas porciones de la red en que el número de usuarios y el tráfico que generan lo justifiquen: esto significa que el ancho de banda sea explotado en su mayoría (idealmente en su totalidad), la mayor parte del tiempo.

Probablemente también serán establecidos canales de acceso vía satélite cuando la topografía del terreno sea muy accidentada, a pesar de la inconveniencia del retardo de la transmisión.

Si los párrafos anteriores causaran la impresión de que el futuro ya casi está aquí, podría surgir la pregunta: ¿entonces ya no habrá más cambios espectaculares en el futuro más lejano? Y la respuesta es: la historia demuestra que cada día surgen nuevas tecnologías que eran inimaginables una o dos generaciones antes, y que éstas se apoyan en todos los conocimientos y la experiencia acumulados a lo largo de la historia, desde las señales de humo y los caracoles usados por indígenas en América, hasta las redes digitales globales, las fibras ópticas y los satélites de comunicación; lo que probablemente suceda, es que estemos perdiendo la capacidad de asombro ante el vertiginoso desarrollo tecnológico de nuestros tiempos.

2.5. Protocolos de comunicación e intercambio de información

Introducción

Ya hemos hablado anteriormente de la necesidad de contar con un protocolo de comunicación para garantizar la integridad, la confidencialidad, la seguridad y algunos otros aspectos de la información que se transmite entre un emisor y un receptor. A continuación, se profundizan algunos aspectos relevantes sobre protocolos de comunicación en una red de información. Recordemos que hablamos de una red digital de información con sistemas como nodos, en los cuales está corriendo una cierta aplicación que requiere comunicarse con otra aplicación que puede estar en otro sistema, en otra red, en otra ciudad, inclusive en otro país en cualquier lugar del mundo; los sistemas están conectados a través de una red, la cual nos garantiza una transmisión confiable gracias a que está controlada por un protocolo de comunicación.

Protocolos de Comunicación

Un protocolo, desde el punto de vista de una red de comunicaciones, es un conjunto de reglas que indican cómo se debe llevar a cabo un intercambio de datos o información. Para que dos o más nodos en una red puedan intercambiar información es necesario que entiendan el mismo protocolo de comunicaciones. Debido a la gran variedad de protocolos, se hizo necesario estandarizarlos y para eso se tomó un diseño estructurado o modular que produjo un modelo jerárquico conocido como *modelo de referencia OSI* (Open Systems Interconnection).

Jerarquías de protocolos

La idea central detrás del modelo es que, para que una aplicación que reside en un nodo A establezca comunicación con una aplicación en un nodo B, debe usar los servicios de una capa de la red; llamemos a ésta, *capa de aplicación*. La capa de aplicación le brinda un conjunto de servicios a las aplicaciones pero a su vez depende de otra capa inferior para trabajar. Llamémosle *capa de transporte de paquetes*. Bajo la capa de transporte residen otras capas con relaciones similares a las ya descritas, hasta llegar a la capa que se encarga del problema del medio físico por el cual viaja finalmente la información de manera electrónica. Llamemos a esta última *capa física*. Esta capa podría encargarse de detectar señales de voltaje en un cable de cobre y agruparlas como unos y ceros para formar un byte, y luego unir los bytes hasta formar una cadena de cierto tamaño predefinido por el protocolo y pasar esa cadena a la capa inmediata superior. Este esquema de *capas* se ilustra más adelante, como el *modelo de referencia OSI*.

Aspectos de diseño

Dependiendo de las funciones y servicios que cada *capa* debe proveer y usar, se deben atacar algunos problemas interesantes al diseñar la arquitectura de red. En primer lugar, dado que en una red existen muchos nodos que quieren comunicarse entre sí, debe existir un mecanismo de direccionamiento que sea capaz de:

- Identificar de manera única a una conexión que parte de un nodo x a un nodo y que está siendo requerida por procesos en dichos nodos.
- Identificar de manera única, para cada conexión, a qué tipo de servicio pertenece.
- Si la interfase soporta mensajes de tamaño restringido, se debe proveer un mecanismo de identificación de mensajes pequeños que pertenecen a uno mayor.
- Identificar en un mismo medio diferentes canales activos simultáneos.
- Debido a que algunos medios físicos no están libres de errores, se debe tener un mecanismo para detectarlos y en alguna de las capas debe haber un mecanismo para corregir el error.
- Ya que los medios no son libres de errores, y si no hay posibilidad de corregir un error, se debe poder pedir la retransmisión de datos.
- Debido a que se puede perder la comunicación entre dos nodos, debe existir un mecanismo para asignar un tiempo máximo de espera en el envío o recepción de datos (time out).
- El protocolo debe contemplar la posibilidad de manejar un medio simplex, half-duplex o full-duplex (esto es: comunicación unidireccional, bidireccional conmutada o bidireccional).

Interfases y servicios

Como ya vimos, cada capa tiene un conjunto de operaciones que realizar y un conjunto de servicios que usa de otra capa. De esta manera identificamos como usuario del servicio a la capa que solicita un servicio y como proveedor a quien la otorga. Cuando una entidad se comunica con otra ubicada en la misma capa pero en diferentes nodos se dice que se establece comunicación entre entidades interlocutoras (peer-entities). Cada capa tiene un conjunto de servicios que ofrecer, el punto exacto donde se puede pedir el servicio se llama punto de acceso al servicio (Service Access Point, SAP). En cada capa, la entidad activa recibe un bloque de datos consistente de un encabezado que tiene significado para el protocolo de esa capa y un cuerpo que contiene datos para ser procesados por esa entidad o que van dirigidos a otra capa.

Relaciones entre servicios y protocolos

Las capas ofrecen servicios de dos tipos generales: orientadas a conexión y no orientadas a conexión y los servicios obtenidos cumplen con cierta calidad de servicio que puede ser un servicio confiable (reliable) o no confiable (non reliable).

Servicios orientados a conexión

Los servicios orientados a conexión se caracterizan porque cumplen tres etapas en su tiempo de vida:

- Etapa 1: Negociación del establecimiento de la conexión.
- Etapa 2: Sesión de intercambio de datos.
- Etapa 3: Negociación del fin de la conexión

Los servicios orientados a conexión pueden ser considerados como *alambrados*, es decir, que existe una conexión física entre los dos interlocutores durante el tiempo de vida de la conexión.

Servicios no orientados a conexión

Los servicios no orientados a conexión carecen de las tres etapas antes descritas y en este caso los interlocutores envían todos los paquetes de datos que componen una parte del diálogo por separado, pudiendo estos llegar a su destino en desorden y por diferentes rutas. Es responsabilidad del destinatario ensamblar los paquetes, pedir retransmisiones de paquetes que se dañaron y darle coherencia al flujo recibido. Los servicios no orientados a conexión se justifican dentro de redes de área local en donde diversos estudios han demostrado que el número de errores es tan pequeño que no vale la pena tener un mecanismo de detección y corrección de los mismos.

Servicios confiables y no confiables

Se dice que un servicio es confiable si nos ofrece una transmisión de datos libre de errores. Para cumplir este requisito, el protocolo debe incluir mecanismos para detectar y/o corregir errores. Existen métodos que únicamente permiten detectar errores (como por ejemplo CRC: Cyclic Reduncy Check) que mediante una fórmula matemática, permiten agregar información extra al mensaje original. Esta información extra es usada del lado del receptor el cual, mediante la misma fórmula, puede corroborar que los datos recibidos son correctos y solicitar una retransmisión de la información en caso contrario. Los métodos de detección de errores son ampliamente usados, ya que son mucho más baratos que implementar un canal de transmisión infalible o un método de detección y corrección de errores.

Existen también mecanismos que permiten además corregir el error. El método más simple de detección y corrección de errores, es agregar redundancia al mensaje. Por ejemplo: si tenemos el mensaje "ABC", se le puede agregar redundancia transmitiendo 3 veces seguidas cada símbolo, de forma que el mensaje transmitido sea algo como "AAABBBCCC".

Conceptos Básicos

Suponiendo que el mensaje recibido por el receptor es: "AUABBGMC", por probabilidades se puede detectar y corregir el error, esto es: si la primera secuencia es "AUA", al haber 2 símbolos idénticos, se puede asumir que es mucho más probable que la secuencia que se quiso transmitir haya sido "AAA". Sin embargo, el método no es infalible. Supongamos que la secuencia recibida sea "AUABBGUDC", entonces los últimos caracteres son "UDC": en este caso no hay suficiente información para asumir cual fue el símbolo que más probablemente se haya querido transmitir. Se pueden tener métodos mucho más sofisticados para la detección y corrección de errores, pero el costo de cómputo y transmisión al hacerlos infalibles es muy elevado.

Como ya se dijo, la corrección de errores puede hacerse con información que está incluida en un paquete dañado o pidiendo su retransmisión al interlocutor. También es común que se incluyan mecanismos para enviar acuses de recibo cuando los paquetes llegan correctamente.

Se dice que un servicio es no confiable si el protocolo no nos asegura que la transmisión está libre de errores y es responsabilidad del protocolo de una capa superior (o de la aplicación) la detección y corrección de errores si esto es pertinente o estadísticamente justificable. A un servicio que es a la vez no orientado a la conexión y no confiable se le conoce como servicio de datagramas (datagram service). Un servicio que es no orientado a la conexión pero que incluye acuse de recibo se le conoce como servicio de datagramas con reconocimiento (acknowledged datagram service). Un tercer tipo de servicio se denomina de petición-respuesta (request-reply) si consiste en un servicio no orientado a conexión y por cada envío de datos se espera una contestación inmediata antes de enviar el siguiente bloque de datos. Este último servicio es útil en el modelo cliente-servidor.

Criptografía.

Del antiguo Egipto a la era digital, los mensajes cifrados han jugado un papel destacado en la Historia. Arma de militares, diplomáticos y espías, son la mejor defensa de las comunicaciones y datos que viajan por Internet. Desde la Antigüedad, el hombre ha hecho gala de su ingenio para garantizar la confidencialidad de sus comunicaciones. La criptografía (del griego *kryptos*, "escondido", y *graphein*, "escribir") es la ciencia de enmascarar los mensajes con signos convencionales, que sólo cobran sentido a la luz de una clave secreta. Sin embargo, hay que señalar que es prácticamente imposible crear un sistema criptográfico infalible. Dicho esto en palabras de un apasionado de la criptografía, Edgar Allan Poe: "es dudoso que el género humano logre crear un enigma que el mismo ingenio humano no resuelva".

La criptografía fue considerada un arte, hasta que C.E. Shannon publicó en 1949 la "Teoría de las comunicaciones secretas", la cual fue aplicada por el NBS (National Bureau of Standards) de E.U. para desarrollar el sistema criptográfico DES (Data Encryption Standard). Entonces la criptografía empezó a ser considerada una ciencia aplicada, debido a su relación con otras ciencias, como la estadística, la teoría de números, la teoría de la información y la teoría de la complejidad computacional.

Este antecedente no ha sido el único: Philip Zimmermann, un criptógrafo aficionado, levantó hace unos años la ira del Gobierno estadounidense. Su delito fue idear un sistema de codificación aparentemente inviolable, el PGP (Pretty Good Privacy), y distribuirlo por las redes de comunicación públicas para que cualquier persona pudiera utilizarlo, algo que no podía agradar a quienes ven en la criptografía un arma de doble filo: útil para los gobiernos y nefasta en manos de terroristas y delincuentes.

Ahora bien, la criptografía corresponde sólo a una parte de la comunicación secreta. Si se requiere secreto para la comunicación, es porque existe desconfianza o peligro de que el mensaje transmitido sea interceptado por alguien a quien no está destinado. Esta persona, si existe, utilizará todos los medios a su alcance para descifrar esos mensajes secretos mediante un conjunto de técnicas y métodos que constituyen una ciencia conocida como criptoanálisis. Al conjunto de ambas ciencias, criptografía y criptoanálisis se le denomina criptología.

En los procesos de almacenamiento y transmisión de la información normalmente aparece el problema de la seguridad. En el almacenamiento, el peligro lo representa el robo del soporte del mensaje o simplemente el acceso no autorizado a esa información, mientras que en las transmisiones lo es la intervención del canal. La protección de la información se lleva a cabo variando su forma. Se llama cifrado (o transformación criptográfica) a una transformación del texto original (llamado también texto inicial o texto claro) que lo convierte en el llamado texto cifrado o criptograma.

Análogamente, se llama descifrado a la transformación que permite recuperar el texto original a partir del texto cifrado. Cada una de estas transformaciones está determinada por un parámetro llamado clave. El conjunto de sus posibles valores se denomina espacio de claves.

Kerckhoffs (s. XIX), en su trabajo titulado "La criptografía militar", recomendó que los sistemas criptográficos cumplieren las siguientes reglas, que efectivamente han sido adoptadas por gran parte de la comunidad criptográfica:

- No debe existir ninguna forma de recuperar mediante el criptograma, el texto inicial o la clave.
- Todo sistema criptográfico debe estar compuesto por dos tipos distintos de información: Pública, como es la familia de algoritmos que lo definen y Privada, como es la clave que se usa en cada cifrado particular. En los sistemas de clave pública, parte de la clave es también información pública.
- La forma de escoger la clave debe ser fácil de recordar y modificar.
- Debe ser factible la comunicación del criptograma por los medios de transmisión habituales.
- La complejidad del proceso de recuperación del texto original debe corresponderse con el beneficio obtenido.

Los sistemas criptográficos que se utilizan actualmente cumplen en general con todas éstas reglas básicas.

Conceptos Básicos

Para el estudio de los sistemas criptográficos es conveniente conocer también la situación del criptoanalista, esta puede ser:

- Conocer sólo el texto cifrado. Esta es la peor situación posible para el criptoanalista, ya que se presenta cuando sólo conoce el criptograma.
- Conocer el texto original. Consiste en que el criptoanalista tiene acceso a una correspondencia de cualquier texto (inicial y cifrado).
- Conocer un texto original escogido. Este caso se da cuando el criptoanalista puede obtener, además del criptograma que trata de descifrar, el cifrado de cualquier texto que él elija, entendiéndose que esto no significa que sepa cifrarlo, sino que lo obtiene ya cifrado.
- Conocer un texto cifrado escogido. Se presenta cuando el criptoanalista puede obtener el texto original correspondiente a determinados textos cifrados de su elección.

Cifrado Simétrico o de Clave Secreta.

Existen sistemas en donde la seguridad radica en la confidencialidad de la clave de cifrado, y de que sea conocida únicamente por emisor y receptor, ya que con esta clave se puede tanto cifrar como descifrar el mensaje.

La principal amenaza criptoanalítica para todos los sistemas criptográficos proviene de la alta redundancia de la fuente. Shannon sugirió por ello dos métodos básicos para frustrar un criptoanálisis estadístico: la difusión y la confusión. El propósito de la difusión consiste en anular la influencia de la redundancia de la fuente sobre el texto cifrado. Hay dos formas de conseguirlo. La primera, conocida como transposición, evita los criptoanálisis basados en las frecuencias de las palabras o caracteres de un texto. La otra manera consiste en hacer que cada letra del texto cifrado dependa de un gran número de letras del texto original. El objetivo de la confusión consiste en hacer que la relación entre la clave y el texto cifrado sea lo más compleja posible, haciendo así que las estadísticas del texto cifrado no estén muy influidas por las del texto original. Eso se consigue normalmente con la técnica de la sustitución. Usadas separadamente, ni confusión ni difusión constituyen buenas técnicas de cifrado.

Se puede hacer otra gran división de los cifrados según el tipo de operación que se realiza. Dada la característica finita del alfabeto y la hipótesis de no variación de la longitud del texto, existen dos opciones para el cifrado: la primera, llamada transposición, consiste en crear el texto cifrado simplemente desordenando las unidades que forman el texto original según una clave. La segunda, llamada sustitución, consiste en sustituir las unidades del texto original por otras según una clave.

Se llama *cifrado producto* al texto resultante de la aplicación iterativa de cifrados sobre textos ya cifrados, es decir, a la composición de varios cifrados. En general, los cifrados simétricos son cifrados producto de las dos operaciones mencionadas, sustitución y transposición. Sustitución y transposición no resultan muy efectivos usados individualmente, sin embargo constituyen la base de sistemas mucho más difíciles de criptoanalizar.

A finales de los años cuarenta, Shannon sugirió nuevas ideas para futuros sistemas de cifrado. Sus sugerencias se referían al uso de operaciones múltiples que mezclaran transposiciones y sustituciones. Estas ideas fueron aprovechadas por IBM en los años setenta, cuando desarrolló un nuevo sistema llamado LUCIFER. Poco después en 1976, el gobierno de EEUU adoptó como estándar un sistema de cifrado basado en el LUCIFER y denominado DES (Data Encryption Standard).

Cifrado asimétrico o de clave pública.

En los cifrados asimétricos o de clave pública la clave de descifrado no se puede calcular a partir de la clave de cifrado. En 1975, dos ingenieros electrónicos de la Universidad de Stanford, Whitfield Diffie y Martin Hellman, sugieren usar problemas computacionalmente irresolubles para el diseño de criptosistemas seguros. La idea consiste básicamente en encontrar un sistema de cifrado computacionalmente fácil (o al menos no tan difícil), de tal manera que el descifrado sea, por el contrario, computacionalmente irresoluble a menos que se conozca la clave. En estos esquemas se utiliza una clave de cifrado (clave pública) y una clave de descifrado (clave secreta o privada) que permite la obtención del texto claro. Cualquier usuario puede cifrar usando la clave pública, pero sólo aquellos que conozcan la clave secreta pueden descifrar correctamente.

En consistencia con el espíritu de la criptografía moderna, y tal cómo sucedía en los sistemas simétricos, los algoritmos de cifrado y de descifrado son públicos, por lo que la seguridad del sistema se basa únicamente en la clave de descifrado.

Existen grandes diferencias en la generación de claves entre el cifrado simétrico y el asimétrico. En los algoritmos simétricos, en los que el conocimiento de la clave de cifrado es equivalente al de la de descifrado, y viceversa, la clave se puede seleccionar de forma aleatoria. Sin embargo, en los algoritmos asimétricos, como existe relación entre clave de cifrado y de descifrado, se necesita un procedimiento para calcular la clave pública a partir de la clave privada que sea computacionalmente eficiente y tal que el cálculo inverso sea imposible de realizar. Por otro lado, hay que resaltar la ventaja que representa en los sistemas asimétricos la posibilidad de iniciar comunicaciones secretas sin haber tenido ningún contacto previo.

A continuación, nombramos algunos de los sistemas de clave pública que han tenido más trascendencia.

- Sistema RSA. Se basa en el hecho de que no existe una forma eficiente de factorizar números que sean productos de dos grandes primos.
- Sistema de Rabin-Williams. Se basa en la factorización del logaritmo discreto.
- Sistema de El Gamal. Se basa en el problema del logaritmo discreto.
- Sistema de Merkle-Hellman. Esta basado en el problema de la mochila.
- Sistema de McEliece. Se basa en la teoría de la codificación algebraica, utilizando el hecho de que la decodificación de un código lineal general es un problema que no se puede resolver eficientemente.

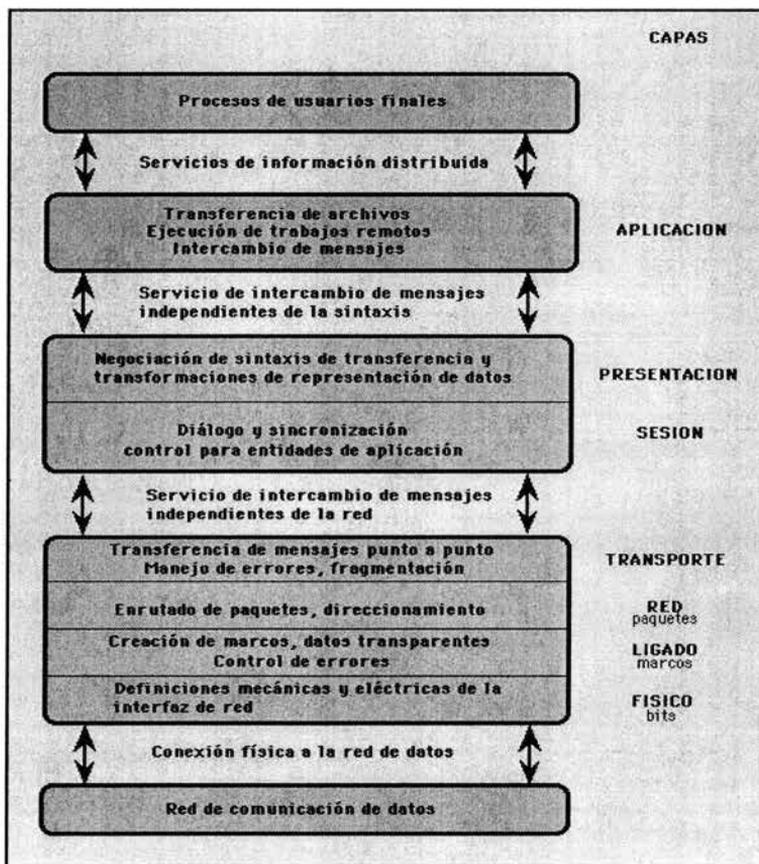
Conceptos Básicos

- Sistemas basados en curvas elípticas. En 1985, la teoría de las curvas elípticas encontró de la mano de Miller su aplicación en la criptografía. Se basa en el hecho de que los cálculos de curvas elípticas en grupos finitos no pueden hacerse eficientemente.
- Sistemas Probabilísticos. La principal diferencia técnica entre el cifrado probabilístico y los criptosistemas de clave pública es que los algoritmos de cifrado son probabilísticos en lugar de determinísticos: el mismo mensaje original puede dar lugar a un gran número de criptogramas distintos. En consecuencia, un criptoanalista que tenga un candidato para el texto original no podría verificar su suposición cifrándolo y comparando el resultado con el criptograma interceptado.

2.6. El modelo de referencia OSI

La Organización de Estándares Internacionales ISO emitió un modelo de referencia para la interconexión de sistemas abiertos (Open Systems Interconnection, OSI). Este modelo tiene la intención de servir como referencia para el diseño de nuevos sistemas de comunicación altamente modulares, flexibles e interconectables.

Capas del modelo de referencia



Esquema del Modelo de Referencia OSI

Capa Física

Se encarga del medio físico por el cual viaja la información, esto es: si el emisor ha transmitido un 1, el receptor debe asegurarse de recibirlo e interpretarlo como un 1. Aquí se estudian aspectos del medio físico que se usa (impedancias, resistencias, atenuaciones, dispersiones, tipo y forma de los conectores, etc.) También se toma en cuenta si el medio permite la comunicación simplex, half-duplex o full-duplex.

Conceptos Básicos

Capa de Ligado

En esta capa, los datos se agrupan en *ventanas* que son transmitidas como una sola unidad. Esta capa se encarga de proporcionar los métodos de detección/corrección de errores, retransmisión y acuses de recibo. En esta capa también se decide cómo acceder al medio físico.

Capa de Red

Esta capa controla el acceso al medio, el enrutado de paquetes, resolución de los cuellos de botella, y en algunos casos la obtención de estadísticas de uso del canal.

Capa de Transporte

La obligación en esta capa es la de tomar datos de la capa de sesión y asegurarse que dichos datos lleguen a su destino. Esta capa se encarga de partir los datos en unidades máximas de transmisión (Maximum Transmission Unit, MTU) en el origen y reensamblarlos en el destino. Sirve también para aislar las capas superiores de los cambios en el hardware que afectan a las capas inferiores. Puede tomar decisiones para optimizar el uso de los canales de transmisión, y proporciona un método de nombrado único para los nodos (y sus diferentes servicios) en la red.

Capa de Sesión

Esta capa ofrece el servicio de establecer sesiones de trabajo entre nodos diferentes de una red. Puede servir para controlar el acceso al canal de transmisión, sincronizar nodos y proporcionar mecanismos que permitan la recuperación de errores o desconexiones a la red.

Capa de Presentación

Esta capa puede aislar la representación de los datos de nuestra aplicación (su interpretación, formato, sintaxis, codificación, etc.) de los usados por las demás capas.

Capa de Aplicación

En esta capa es donde se encapsulan, utilizan, interpretan, procesan y despliegan los datos relativos a la aplicación.

Transmisión de datos en el modelo OSI

Un envío de datos típico bajo el modelo de referencia OSI comienza con una aplicación P en un nodo cualquiera de la red que desea enviar información a un punto D. Los datos en P se van *encapsulando*, a partir de la capa de aplicación, siendo pasados a las capas inferiores, hasta llegar a la física, donde se realiza la transmisión hasta el nodo D. En el nodo D, los datos se van *desencapsulando* y pasando a capas superiores, donde finalmente son entregados por la capa de aplicación a nuestro programa que desea utilizar la información recibida.

De manera virtual, se establecen conexiones directas entre las capas del mismo nombre de los dos diferentes nodos. Por ejemplo, el paquete que envía la capa de red es interpretado por la capa de red en el destino y no por otra capa. Para las capas inferiores de la de red, dicho paquete fue interpretado como datos, y para las capas superiores (transporte, sesión, presentación y aplicación) como un paquete compuesto de datos y encabezado. Por otro lado, todas las capas, excepto la de aplicación, procesan los paquetes realizando operaciones que sólo sirven para verificar que el paquete de datos real esté íntegro o para que éste llegue a su destino, sin que los datos por sí mismos sufran algún cambio

2.7. Java

2.7.1. Historia de java

Introducción: El nacimiento de Java

Java es un lenguaje de programación de Sun Microsystems originalmente llamado *Oak*, que fue concebido bajo la dirección de James Gosling y Bill Joy, quienes pertenecían a una subsidiaria de Sun, conocida como *FirstPerson Inc.* Oak nació para programar pequeños dispositivos electrodomésticos, como los asistentes personales digitales PDA's (*Personal Digital Assistants*) y un poco más adelante se utilizó para ejecutar aplicaciones para televisión interactiva. Ninguno de estos productos tuvo éxito comercial en ese momento. Gosling y Joy se quedaron con una tecnología robusta, eficiente, orientada a objetos e independiente de la arquitectura, pero hasta entonces, sin ninguna utilidad práctica.

No pasó mucho tiempo, para que Sun se diera cuenta de que todas estas características cubrían a la perfección las necesidades de las aplicaciones de Internet. De esta manera, con unos cuantos retoques, Oak se convirtió en Java. Además de todas las características que posee Java (modelo de objetos dinámico, sistema estricto de tipos, paquetes, hilos, excepciones, etcétera), cuando Netscape Inc. anunció su incorporación dentro de su navegador (*Netscape Navigator*), el nivel de interés sobre el lenguaje creció dramáticamente, debido al importante número de personas que utilizan WWW diariamente. Todo lo anterior se ha conjugado para lograr el éxito actual de Java, siendo el actor principal su máquina virtual.

Las principales características de Java

Java es un lenguaje de programación que está soportado por dos elementos fundamentales: el compilador y la máquina virtual. El compilador, encargado de traducir los programas a un formato capaz de ser entendido por la computadora que debe ejecutarlo, es un elemento común con cualquier lenguaje de programación. La máquina virtual Java (*la máquina encargada de ejecutar los programas traducidos por el compilador*) aporta nuevas características al modelo de programación del lenguaje. Utilizando como fuente uno de los primeros artículos sobre Java, escrito por su creador, podemos decir que las principales características de Java son que se trata de un lenguaje: simple, orientado a objetos, distribuido, robusto, seguro, arquitecturalmente neutro, portable, concurrente, dinámico, interpretado y de alto rendimiento. Veamos ahora qué es lo que hay detrás de cada una de estas características.

Simple

Java ofrece toda la funcionalidad de lenguajes potentes como C y C++ (en los que está basado), pero sin las características menos usadas y más confusas de éstos. De hecho, dado que C y C++ son lenguajes enormemente difundidos, la sintaxis de Java es muy similar, con el objetivo de facilitar su aprendizaje. Las características que Java no incluye de C y C++ son: manejo explícito de la memoria dinámica y la aritmética de punteros, registros (*struct*), redefinición de tipos (*typedef*) y macros (*#define*). Según los creadores del lenguaje, esto reduce en un 50% los errores de programación más comunes. Además, Java es compacto: una máquina virtual ocupa menos de 0.5MB de memoria.

Orientado a objetos

Java implementa la tecnología básica de C++ con algunas mejoras y elimina algunas características para mantener el objetivo de simplicidad del lenguaje. Java trabaja con sus datos como objetos y con interfaces a esos objetos.

Distribuido

Java se ha construido con extensas capacidades de interconexión a través de redes, que forman parte de sus componentes básicos. El modelo de acceso a la información a través de la red es en todo equivalente al acceso a la información disponible localmente.

Robusto

Java realiza verificaciones en busca de potenciales problemas tanto en tiempo de compilación como en tiempo de ejecución. La comprobación de tipos en Java ayuda a detectar errores lo antes posible en el ciclo de desarrollo.

El manejo de memoria es realizado internamente, sin que el programador deba reservarla o liberarla explícitamente, eliminando así problemas de corrupción de datos y accesos no autorizados. Siguiendo esta filosofía, los *arreglos* (o *arrays*) constituyen un tipo específico del lenguaje.

Seguro

La seguridad en Java tiene dos facetas. En el lenguaje, aspectos como los punteros se eliminan para prevenir el acceso ilegal a la memoria. Por otro lado, el código Java pasa muchas comprobaciones antes de ser ejecutado por la máquina virtual. El código pasa a través de un verificador de *bytecodes* que comprueba el formato de los fragmentos de código y aplica un probador de teoremas para detectar fragmentos de código ilegal, como por ejemplo accesos no autorizados a las zonas de memoria de los objetos o intentos de cambio del tipo de algún objeto. Si los *bytecodes* pasan la verificación sin producir un mensaje de error, la máquina virtual se ha asegurado de que el código: no produce desbordamientos de operandos en la pila; el tipo de los parámetros de todos los códigos de operación son conocidos; no ha ocurrido ninguna conversión ilegal de datos y el acceso a los campos de los objetos es legal.

El cargador de clases es el otro elemento que ayuda a mantener la seguridad en la máquina virtual Java, separando el espacio de nombres del sistema local del de los recursos procedentes de la red. Esto limita cualquier aplicación del tipo Caballo de Troya ya que las clases se buscan primero entre las locales y luego entre las procedentes del exterior. Las clases importadas de la red se almacenan en un espacio de nombres privado, asociado con el origen. Cuando una clase del espacio de nombres privado accede a otra clase primero se busca entre las clases predefinidas del sistema local y luego en el espacio de nombres de la clase que hace la referencia. Así se evita que las clases puedan ser suplantadas.

Conceptos Básicos

Arquitecturalmente neutro

El código objeto producido por el compilador de Java tiene un formato independiente de la arquitectura (sistema operativo, procesador, etc.) de la máquina donde se ejecuta. Como ya hemos visto, este es código ejecutable (bytecode) en una máquina virtual que es implementada por el sistema de ejecución (run-time), que es el que tiene en consideración las particularidades de la máquina donde el programa es ejecutado.

Portable

Más allá de la portabilidad básica por ser independiente de la arquitectura, Java implementa otros estándares de portabilidad para facilitar el desarrollo. Por ejemplo, los enteros son siempre de 32 bits y almacenados en complemento a 2 con formato BigEndian, mientras que las cadenas de caracteres utilizan Unicode. Además, la interfase de ventanas de Java (AWT) es un sistema abstracto de ventanas que permite su implementación en diferentes entornos, como X-Windows, MS-Windows, MacOS, OS/2, etc.

Concurrente

Java permite que múltiples flujos de control (threads) puedan ser ejecutados por el mismo programa. Los threads son básicamente pequeños procesos o piezas independientes de un gran proceso. Al estar los threads construidos en el lenguaje, son más fáciles de usar, más robustos y, por supuesto, más portables que sus homólogos en otros lenguajes que no los soportan de manera nativa. La capacidad de programación concurrente permite mejorar el rendimiento interactivo y el comportamiento en tiempo real.

Dinámico

Java se beneficia en todo lo posible de la tecnología orientada a objetos. Los módulos que componen la aplicación no son conectados hasta el tiempo de ejecución. Las librerías nuevas o actualizadas son incorporadas (siempre que mantengan la interfase) automáticamente. Por otro lado, Java proporciona mecanismos para cargar dinámicamente clases desde la red, de manera que nuevos contenidos de información podrán ser tratados por manejadores específicos.

Interpretado

Dado que la máquina virtual Java es un programa que se ejecuta sobre el sistema operativo de la computadora, la ejecución de un programa Java se basa, en esencia, en un intérprete (la máquina virtual), aunque no se trata de un intérprete en el sentido tradicional: los bytecodes Java han pasado ya por las etapas de validación del compilador.

De alto rendimiento

Para los casos en que la velocidad del intérprete Java no resulte suficiente, se dispone ya de mecanismos como los compiladores JIT (Just In Time), que se encargan de traducir (a medida que va siendo necesario) los bytecodes Java a instrucciones de código nativo de la máquina en donde se ejecuta el programa. Otros mecanismos, como compiladores incrementales y sistemas dedicados para tiempo real, se encuentran también en el mercado.

Las Plataformas de Java

Sun ha agrupado sus tecnologías Java en tres ediciones:

Java 2 Platform, Standard Edition (J2SE): Constituye la base de la tecnología Java para el desarrollo de aplicaciones de propósito general.

Java 2 Platform, Enterprise Edition (J2EE): Simplifica el desarrollo de aplicaciones empresariales mediante la utilización de diversos componentes (EJB's, servlets, JSP's, etc.), proporcionando un conjunto completo de servicios para dichos componentes y manejando automáticamente muchos detalles del comportamiento de la aplicación.

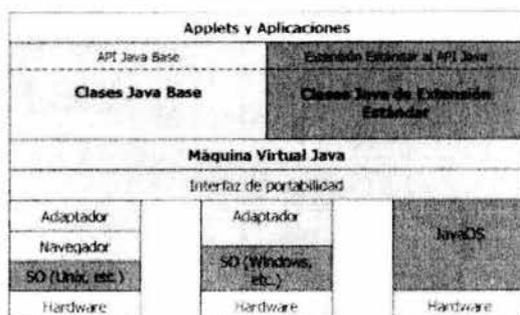
Java 2 Platform, Micro Edition (J2ME): Dirigida específicamente al mercado de consumo, que cubre un rango de productos portátiles pequeños como, smart cards, teléfonos móviles, PDA's, etc.

2.7.2. Aspectos Técnicos de Java

La Máquina Virtual Java (MVJ)

La Máquina Virtual Java es el núcleo del lenguaje de programación. De hecho, es imposible ejecutar un programa Java sin ejecutar alguna implementación de la MVJ. En la MVJ se encuentra el motor que en realidad ejecuta el programa y es la clave de muchas de las características principales del lenguaje, como la portabilidad, la eficiencia y la seguridad. Siempre que se ejecuta un programa Java, las instrucciones que lo componen no son ejecutadas directamente por el hardware sobre el que subyace, sino que son pasadas a un elemento de software intermedio, que es el encargado de que las instrucciones sean ejecutadas por el hardware. Es decir, el código Java no se ejecuta directamente sobre un procesador físico, sino sobre un procesador virtual Java.

La representación de los códigos de instrucción Java (*bytecode*) es simbólica, en el sentido de que los desplazamientos e índices dentro de los métodos no son constantes, sino que son cadenas de caracteres o nombres simbólicos. Estos nombres son resueltos la primera vez que se ejecuta el método, es decir, el nombre simbólico se busca dentro del archivo de clase (*.class*) y se determina el valor numérico del desplazamiento. Este valor es guardado para aumentar la velocidad de futuros accesos. Gracias a esto, es posible introducir un nuevo método o sobrescribir uno existente en tiempo de ejecución, sin afectar o romper la estructura del código. En la siguiente figura puede observarse la capa de software que implementa a la máquina virtual Java. Esta capa de software oculta los detalles inherentes de la plataforma a las aplicaciones Java que se ejecuten sobre ella. Debido a que la plataforma Java fue diseñada pensando en que se implementaría sobre una amplia gama de sistemas operativos y de procesadores, se incluyeron dos capas de software para aumentar su portabilidad. La primera dependiente de la plataforma es llamada *adaptador*, mientras que la segunda, que es independiente de la plataforma, se le llama *interfaz de portabilidad*. De esta manera, la única parte que se tiene que escribir para una plataforma nueva, es el adaptador. El sistema operativo proporciona los servicios de manejo de ventanas, red, sistema de archivos, etcétera.

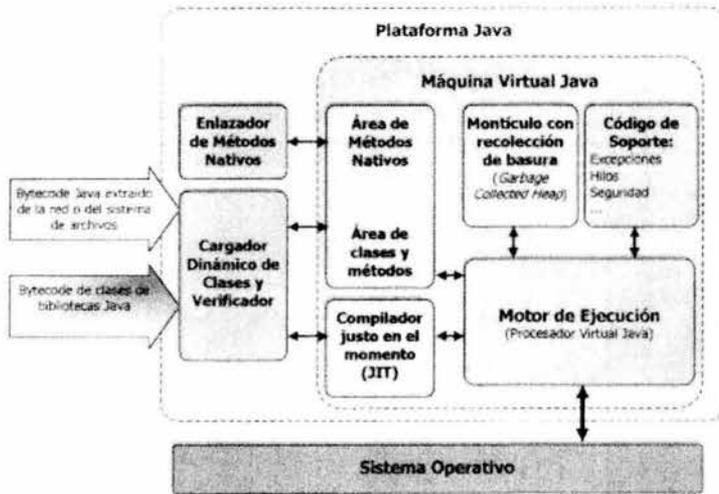


La plataforma Java (Sistema en tiempo de ejecución)

Sun Microsystems utiliza el término *Máquina Virtual Java*, para referirse a la especificación abstracta de una máquina de software para ejecutar los programas. La especificación de esta máquina virtual, define elementos como el formato de los archivos de clases (.class), así como la semántica de cada una de las instrucciones que componen el conjunto de instrucciones de la máquina virtual. A las implantaciones de esta especificación se les conocen como *Sistemas en Tiempo de Ejecución Java*.

Sistema en Tiempo de Ejecución

En la figura se puede observar los componentes típicos de un sistema de tiempo de ejecución. Ejemplos de Sistemas de tiempo de ejecución son el Navegador de Netscape, el Internet Explorer de Microsoft y el programa Java (incluido en el JDK). Un sistema de tiempo de ejecución podría representarse de la siguiente forma:



Entorno de Ejecución Java (Java Runtime Environment, JRE)

- Motor de ejecución. El procesador virtual que se encarga de ejecutar el código (bytecode), generado por algún compilador Java.
- Manejador de memoria. Encargado de obtener memoria para las nuevas instancias de objetos, arreglos, etcétera, así como realizar tareas de recolección de basura.
- Manejador de errores y excepciones. Encargado de generar, lanzar y atrapar excepciones.
- Soporte de métodos nativos. Encargado de llamar métodos de C++ o funciones de C desde métodos Java y viceversa.
- Interfaz multihilos. Encargada de proporcionar el soporte para hilos y monitores.

Conceptos Básicos

- Cargador de clases. Su función es cargar dinámicamente las clases Java a partir de los archivos de clase (.class).
- Administrador de seguridad. Se encarga de asegurar que las clases cargadas sean seguras, así como controlar el acceso a los recursos del sistema.

Adicionalmente, existe un conjunto de clases Java estándar, fuertemente ligadas a la implementación de cada MVJ en particular. Ejemplos de esto los tenemos en las clases encargadas de los accesos a los recursos de la red, manejar el sistema de ventanas, los hilos y el sistema de archivos local. Todos estos elementos en conjunto actúan como una interfaz de alto nivel, para acceder a los recursos del sistema operativo. Esta interfaz es la clave de la portabilidad de los programas Java, debido a que independientemente del hardware o sistema operativo sobre el que se esté trabajando, la máquina virtual Java oculta todas estas diferencias.

A continuación describiremos con mayor detalle cada uno de estos elementos.

Motor de Ejecución

Es la entidad de hardware o software, que ejecuta las instrucciones contenidas en los códigos de operación (*bytecodes*) que implementan los métodos Java.

El Conjunto de Instrucciones del Procesador Virtual

Muchas de las instrucciones del procesador virtual Java, son muy similares a las que se pueden encontrar para los procesadores comunes y corrientes, como los Intel, es decir, incluyen los grupos de instrucciones típicos como los aritméticos, los de control de flujo, de acceso a memoria, a la pila, etcétera. Una de las características más significativas del conjunto de instrucciones del procesador virtual Java, es que están basadas en la pila y utilizan *posiciones de memoria numeradas*, en lugar de registros.

El Verificador de Java

Cuando los desarrolladores de Java estaban diseñando el conjunto de instrucciones para la máquina virtual, tenían dos metas en mente. La primera era que el conjunto de instrucciones fuera similar a las instrucciones que se pueden encontrar en los procesadores reales. La segunda era construir un conjunto de instrucciones que fuera fácilmente verificable.

En Java, justo después de que se obtiene una clase del sistema de archivos o de Internet, la máquina virtual puede ejecutar un verificador que se encargue precisamente de constatar que la estructura del archivo de clase es correcta. El verificador se asegura que el archivo tenga el número mágico (0xCAFEBABE) y que todos los registros que contiene el archivo tengan la longitud correcta, pero aún más importante, comprueba que todos los códigos de operación sean seguros de ejecutar.

Administrador de Memoria

Java utiliza un modelo de memoria conocido como *administración automática del almacenamiento* (*automatic storage management*), en el que el sistema en tiempo de ejecución de Java mantiene un seguimiento de los objetos. En el momento que no están siendo referenciados por alguien, el recolector de basura automáticamente libera la memoria asociada con ellos. Existen muchas maneras de implementar recolectores de basura, pero sólo mencionaremos los 2 algoritmos más comúnmente implementados:

- *Contabilizar referencias*. La máquina virtual Java asocia un contador a cada instancia de un objeto, donde se refleja el número de referencias hacia él. Cuando este contador es 0, la memoria asociada al objeto es susceptible de ser liberada. Aún cuando este algoritmo es muy sencillo y de bajo costo en términos computacionales, presenta problemas con estructuras de datos circulares.
- *Marcar e intercambiar (Mark-and-Sweep)*. Este es el esquema más común para implementar el manejo de almacenamiento automático. Consiste en almacenar los objetos en un montículo (heap) de un tamaño considerable y marcar periódicamente (generalmente mediante un bit en un campo que se utiliza para este fin) los objetos que no tengan ninguna referencia hacia ellos. Adicionalmente existe un montículo alterno, donde los objetos que no han sido marcados, son movidos periódicamente. Una vez en el montículo alterno, el recolector de basura se encarga de actualizar las referencias de los objetos a sus nuevas localidades. De esta manera se genera un nuevo montículo, que contiene únicamente objetos que están siendo utilizados.

La gran mayoría de las implementaciones de máquinas virtuales Java utilizan el algoritmo de *Marcar e Intercambiar*.

Administrador de Errores y Excepciones

Las excepciones son la manera como Java indica que ha ocurrido algo *extraño* durante la ejecución de un programa. Todas las excepciones en Java son instancias de la clase *java.lang.Throwable* o de alguna otra que la especialice. Las clases *java.lang.Exception* y *java.lang.Error*, heredan directamente de la clase *java.lang.Throwable*. Cuando se genera una excepción, el sistema de tiempo de ejecución de Java, y en particular el manejador (*handler*) de errores y excepciones, busca un manejador para esa excepción, comenzando por el método que la originó y después hacia abajo en la pila de llamadas. En el caso que no se encuentre un manejador para alguna excepción previamente lanzada, se ejecuta el manejador del sistema, cuya acción típica es imprimir un mensaje de error y terminar la ejecución del programa.

Soporte para Métodos Nativos

Las clases en Java pueden contener métodos que no estén implementados por códigos de operación (*bytecode*) Java, sino por algún otro lenguaje compilado en código nativo y almacenado en bibliotecas de enlace dinámico, como las DLL de Windows o las bibliotecas compartidas SO de Solaris. El sistema de tiempo de ejecución incluye el código necesario para cargar dinámicamente y ejecutar el código nativo que implementa estos métodos.

Interfaz de Hilos

Java es un lenguaje que permite la ejecución concurrente de varios hilos de ejecución, es decir, el sistema de tiempo de ejecución de Java tiene la posibilidad de crear más de un procesador virtual Java, donde se pueden ejecutar diferentes flujos de instrucciones, cada uno con su propia pila y su propio estado local. En el conjunto de instrucciones de la máquina virtual Java, sólo existen dos directamente relacionadas con los hilos, *monitorenter* y *monitorexit*, que sirven para definir secciones de código, que deben ejecutarse en exclusión mutua. El resto del soporte de los hilos se realiza atrapando llamadas a los métodos pertenecientes a la clase `java.lang.Thread`.

Cargador de Clases

Los programas Java están completamente estructurados en clases. Por lo tanto, una función muy importante del sistema en tiempo de ejecución, es cargar, enlazar e inicializar clases dinámicamente, de forma que sea posible instalar componentes de software en tiempo de ejecución. El proceso de cargado de las clases se realiza sobre demanda, hasta el último momento posible. La Máquina Virtual Java utiliza dos mecanismos para cargar las clases. El primero consiste en un cargador de clases del sistema, cuya función es cargar todas las clases estándar de Java, así como la clase cuyo nombre es solicitada vía la línea de comandos. De manera adicional, existe un segundo mecanismo para cargar clases dentro del sistema, utilizando dentro de un programa a una instancia de la clase `java.lang.ClassLoader` o alguna otra definida por el usuario que la especialice.

Arquitectura de Seguridad en Java

Java utiliza una serie de mecanismos de seguridad, con el fin de dificultar la escritura de programas maliciosos que pudiesen afectar la integridad de las aplicaciones y los datos de los usuarios. Cada sistema en tiempo de ejecución Java tiene la capacidad de definir sus propias políticas de seguridad, mediante la implementación de un *administrador de seguridad* (*security manager*), cuya función es proteger al sistema de tiempo de ejecución, definiendo el ámbito de cada programa Java en cuanto a las capacidades de acceder a ciertos recursos. El modelo de seguridad original proporcionado por la plataforma Java, es conocido como la *caja de arena* (*sandbox*), que consiste en proporcionar un ambiente de ejecución muy restrictivo para código no confiable que haya sido obtenido de la red. Como hemos mencionado, la máquina implementa otros mecanismos de seguridad, desde el nivel de lenguaje de programación, como la verificación estricta de tipos de datos, manejo automático de la memoria, recolección automática de basura, verificación de los límites de las cadenas y arreglos, etcétera. Finalmente, el acceso a los recursos importantes del sistema, es administrado entre el sistema de tiempo de ejecución y el administrador de seguridad (*Security Manager*), que es implementado por la clase `java.lang.SecurityManager`, que permite a las MVJ incorporar políticas de seguridad. De esta manera, es posible para las aplicaciones determinar si una operación es insegura o contraviene las políticas de seguridad, antes de ejecutarla. El JDK 1.1 introduce el concepto de *applet firmado* (*signed applet*), en el que los *applets* que poseen una firma digital correcta, son considerados como confiables y pueden correr fuera de una *sandbox*.

Desventajas de las Máquinas Virtuales

Una de las razones por que las máquinas virtuales no son la panacea de la computación, es que agregan gran complejidad al sistema en tiempo de ejecución, y están basados en la suposición de que la plataforma sobre la que subyacen, cumple con diversos *estándares*: algo que no siempre ocurre.

Deficiencias de la MVJ

- Conjunto de instrucciones no ortogonal. Un lenguaje de programación es ortogonal, si tiene el mismo número de instrucciones asociadas a cada uno de los tipos de datos.
- Difícil de extender. Debido a que se utiliza un byte para codificar el código de operación de las instrucciones del procesador virtual Java (de ahí el nombre de bytecode), es difícil agregar nuevas instrucciones.
- No posee un árbol de análisis sintáctico. El código intermedio, usado en la MVJ, es simple y plano, es decir: no incluye información acerca de la estructura del método original. En un lenguaje completamente compilado, esta información juega un papel muy importante en la optimización, debido a que permite trabajar con el control de dependencias y de flujo.

2.7.3. J2EE

J2EE proporciona una arquitectura de aplicaciones multicapa basada en componentes desplegados por un servidor. Las API's de Java Enterprise Edition proporcionan interfaces de programación para diversas implementaciones *middleware* como directorios, seguridad, transacciones, mensajes, Bases de Datos y comunicaciones remotas. Los EJB's son la piedra angular de estas API's y forman parte de una plataforma que representa un estándar para implementar y desplegar aplicaciones empresariales. J2EE ha tenido una evolución natural para resolver las necesidades de desarrollo en tecnologías de información basadas en servidor, proporciona soporte para el desarrollo de aplicaciones multicapa, cuenta con un entorno de ejecución consistente e integrado y permite la portabilidad e interoperabilidad de aplicaciones

El servidor J2EE (1.3.1) soporta las siguientes API's y tecnologías:

HTTP y HTTPS, COS naming, Enterprise JavaBeans (EJB) 2.0, Java Servlets 2.3, JavaServer Pages (JSP) 1.2, Java Messaging Service (JMS) 1.0.2, J2EE Connector 1.0, JDBC Standard Extension 2.0, Java Transaction API (JTA) 1.0, JavaMail 1.2 y Java API for XML Parsing (JAXP) 1.1

El runtime de J2EE consta de contenedores y componentes. Estos últimos son piezas de código de aplicación que pueden ser ensambladas en aplicaciones y corren dentro de un entorno llamado contenedor

Arquitectura J2EE

Un contenedor de aplicaciones J2EE proporciona API's (J2SE y J2EE) y el entorno de ejecución necesarios para la ejecución de componentes J2EE.

Contenedores y Componentes

Los contenedores son responsables de manejar la Concurrencia, Seguridad, Disponibilidad, Escalabilidad, Transacciones y Distribución.

Los componentes deben encargarse de la Presentación, la lógica del negocio y opcionalmente a la lógica Acceso a Datos.

Especificación de la plataforma J2EE

La plataforma J2EE especifica un contenedor para cada tipo de componente. Cada tipo de contenedor debe soportar un conjunto de servicios definidos como estándar en la especificación.

Los applets son componentes GUI que normalmente se ejecutan en un Navegador, pueden ejecutarse en otras aplicaciones o dispositivos que soporten el modelo de programación de applets y son utilizados normalmente para proporcionar una interfase de usuario a aplicaciones J2EE.

Las Aplicaciones Cliente son programas desarrollados en Java que normalmente incorporan un GUI y tienen acceso a todos los servicios del servidor J2EE.

Los Servlets son programas Java que extienden la funcionalidad de un servidor Web, generando contenido dinámico e interactuando con clientes Web usando el paradigma Petición-Respuesta (Request-Response). Son objetos basados en el API *Servlet*, son manejados y mapeados a URL's por un contenedor con una estructura simple. Son soportados en los principales Servidores Web y Servidores de Aplicaciones, independientemente de la plataforma y del servidor. Los Servlets brindan algunas ventajas con respecto a otras arquitecturas: no tienen las limitaciones de los CGI's; son soportados ampliamente por herramientas de terceros y servidores Web; tienen acceso a toda la familia de API's de Java; son fiables, rápidos, escalables, independientes de la plataforma y del servidor, además de que la mayoría de los servidores permiten la recarga de aplicaciones mediante acciones de administración.

Los componentes conocidos como JSP's se refieren a una tecnología Web extensible que utiliza datos, elementos, lenguaje script y objetos Java para devolver contenido dinámico al cliente. Los JSP's son documentos HTML con etiquetas especiales y código Java que, en tiempo de ejecución, es analizado y compilado en un servlet. Permiten la separación efectiva del código HTML y la lógica de negocio en las páginas Web, además proporcionan un mecanismo más simple que los servlets para crear contenido dinámico. Los beneficios que los JSP's brindan son: permite la Separación de contenido y lógica de visualización; un desarrollo más simple con JSP, JavaBeans y etiquetas personalizadas (custom tags); soportan la reutilización mediante el uso de componentes; su compilación es automática cuando se cambia el código fuente; son fáciles de escribir e independientes de la plataforma.

Comparación Servlets y JSP's

Los Servlets son archivos de código HTML en Java que aceptan cualquier forma de datos aunque no son tan fáciles de escribir. Los JSP's son código parecido a Java en HTML con texto estructurado, fáciles de escribir y cuyo código se compila en un servlet.

Enterprise Java Beans

Los EJB (Enterprise Java Beans) son componentes de servidor desarrollados de acuerdo con la especificación EJB: objetos distribuidos que corren dentro de un contenedor EJB, que proporcionan servicios remotos a clientes en la red. Existen 2 tipos de Enterprise Java Beans: Session Beans (con estado y sin estado) y Entity Beans (con persistencia manejada por contenedor y manejada por bean).

Un session bean representa comportamientos asociados con sesiones de cliente: cliente único, vida corta, no sobreviven a la aplicación. Los session beans sin estado son útiles para realizar transacciones en un solo paso. Los session beans con estado son mejores cuando se requieren de múltiples pasos para realizar la transacción

Los Entity bean representan colecciones de datos y encapsulan operaciones sobre los datos que representan: múltiples clientes, vida larga, pueden sobrevivir a la aplicación. Un Bean Managed Persistence bean (BMP) contiene el código para almacenar y recuperar datos. Un Container Managed Persistence bean (CMP) permite al contenedor manejar todos los aspectos de acceso a los datos

Conceptos Básicos

Servicios

Existe un conjunto de servicios que se especifican dentro de la plataforma J2EE en forma de una API estándar:

COS Naming (Common Object Services)/JNDI (Java Naming and Directory Interface). Diseñada para estandarizar el acceso a servicios de nombres y de directorio. COS/JNDI proporcionan un mecanismo simple para que componentes J2EE puedan acceder a otros objetos que requieran.

JDBC. El API JDBC permite el acceso a los datos de Bases de Datos relacionales y otros repositorios. El API de JDBC incluye manejo de SQL User-Defined Types (UDT's), manipulación de rowsets, connection pooling y soporte de transacciones distribuidas.

JavaMail. Para soportar aplicaciones como websites de comercio electrónico. Proporciona la posibilidad de enviar confirmaciones, acuses de recibo, etc.

Conformidad CORBA. J2EE soporta 2 tecnologías CORBA: JavaIDL, que permite a aplicaciones Java interactuar con cualquier sistema empresarial conforme con CORBA y RMI-IIOP: Combina RMI (Java Remote Method Invocation) con IIOP (Internet Inter-ORB Protocol) de CORBA para una integración más sencilla entre aplicaciones J2EE y otras aplicaciones ya existentes.

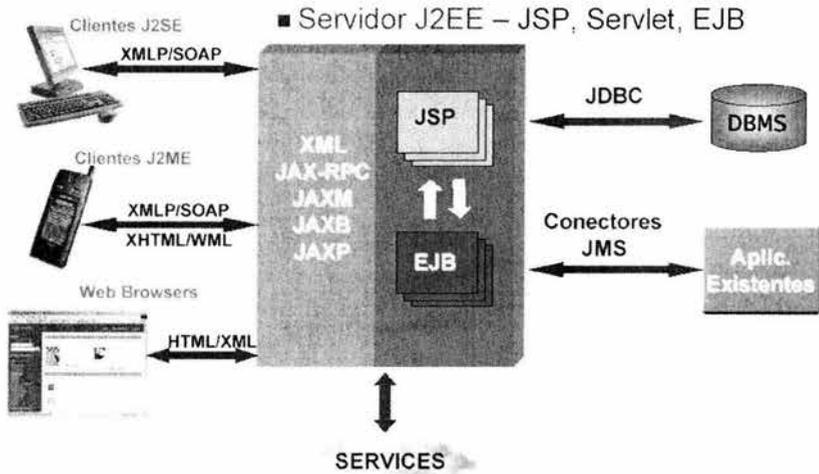
JTA (Java Transaction API). Mientras J2EE proporciona soporte automático para transacciones, JTA proporciona un modo para que clientes y componentes J2EE gestionen sus propias transacciones y para que múltiples componentes participen en una única transacción.

XML Deployment Descriptors. J2EE define un set de descriptores en XML. Con XML, J2EE simplifica el desarrollo de componentes y aplicaciones.

JMS (Java Messaging Service). JMS define mecanismos estándar para que componentes puedan enviar y recibir mensajes de forma asíncrona.

JAXP (Java Api for XML Parsing). Proporciona soporte para el análisis sintáctico de archivos con formato XML.

En la siguiente figura, se ilustra un ejemplo de un modelo basado en *Web Services* que se puede implementar utilizando la plataforma J2EE; más adelante se describirán detalladamente las características de esta tecnología.



Plataforma J2EE / Web Services

76

2.7.4. J2ME

El programador Java que tiene algo de experiencia ha utilizado J2SE, Java 2 Standard Edition. J2SE es el conjunto básico de herramientas usadas para desarrollar applets y aplicaciones *standalone* Java. Sun Microsystems no se planteó hacer un conjunto de herramientas estándar hasta la llegada de Java 2, pues coincidió con los planes de expansión del lenguaje para software empresarial. El software empresarial tiene unas características propias muy marcadas: está pensado no para ser ejecutado en un equipo, sino para ejecutarse sobre una red de computadoras de manera distribuida y remota. De hecho, el sistema se monta sobre varias unidades o aplicaciones. En muchos casos, el software empresarial requiere ser capaz de integrar datos provenientes de entornos heterogéneos.

Para este entorno, por sus exigencias y características, Sun ha diseñado J2EE, Java 2 Enterprise Edition. Sun separó J2SE de J2EE porque este último exigía características muy pesadas o especializadas de I/O, trabajo en red, etc. Finalmente y por razones de eficiencia separó ambos productos. Hoy J2EE es un superconjunto de J2SE pues contiene toda la funcionalidad de este y muchas características adicionales.

Sun ha separado J2ME, Java 2 Micro Edition por las mismas razones. Los dispositivos inalámbricos tienen menos potencia y mucha menor capacidad gráfica que los PC de escritorio. Por ello, J2ME representa una versión simplificada de J2SE pensada para dispositivos con estas limitaciones. El conjunto de J2ME, J2SE y J2EE le llamamos tecnología Java 2. Dicho esto, es importante señalar que J2ME tiene la característica de tener una parte de su API fija, es decir, aplicable a todos los dispositivos inalámbricos y una parte que es específica para ciertos dispositivos; como por ejemplo: la API específica de Palm y la de teléfonos móviles son evidentemente distintas.

Posteriormente analizaremos más a fondo la plataforma J2ME y sus características principales.

3.

Análisis

ESTA TESIS NO SALE
DE LA BIBLIOTECA

3.1. Nuevas Tecnologías Para la Integración de Sistemas de Información.

3.1.1. Estandarización

Introducción

"Los estándares hacen que la vida sea más simple, permitiendo mayor fiabilidad y efectividad en los bienes y servicios que usamos"

¿Alguna vez usted se ha preguntado por qué su tarjeta de crédito puede insertarse en cualquier cajero automático del mundo? Es porque los fabricantes del plástico de las tarjetas se basan en un estándar conocido como ISO 7810 en donde se definen las dimensiones del plástico (85mm de largo, 54mm ancho y 0.8mm de grosor); en otro estándar, el ISO 7811, se definen las características del grabado de la banda magnética de la tarjeta.

Este es sólo un ejemplo de los miles de estándares que existen hoy en día, los cuales rigen nuestras actividades diarias, pues regulan los bienes y servicios que empleamos, permitiendo olvidarnos de marcas o de algún fabricante en especial. La estandarización evita las arquitecturas cerradas, los monopolios y los esquemas propietarios. Así mismo, los estándares son la base de los productos y típicamente son los que marcan la diferencia entre la comunicación y la incompatibilidad.

La historia de la estandarización

A principios del siglo XIX Europa vivía en un estado de agitación; los efectos de la revolución industrial se hacían evidentes en cualquier parte del continente. La revolución de la transportación dio inicio con la aparición de la máquina de vapor y el ferrocarril. Los rieles por los que los trenes se desplazaban fue el primer problema de estandarización entre los países; éstos tenían que ponerse de acuerdo en las dimensiones, material y las demás características de las vías por donde pasaría el tren. Tal situación de entendimiento fue la ideal para la introducción del telégrafo. Al ponerse en funcionamiento este nuevo medio de comunicación, inmediatamente se hicieron evidentes sus beneficios, acercando aún más a las empresas e industrias que existían en ese tiempo y quienes tenían una imperiosa necesidad de difundir noticias y mensajes de manera rápida y eficiente. Tanto el ferrocarril como el telégrafo transformaron de manera notable a la Europa del Siglo XIX.

Con el propósito de buscar una estructura y un método de funcionamiento que permitieran conocer los problemas planteados por las nuevas tecnologías de comunicación, así como también las demandas de los usuarios, en 1865 se fundó la Unión Internacional de Telegrafía (ITU, por sus siglas en inglés). La ITU fue la primera organización intergubernamental e internacional que se creó. Sin lugar a duda, la ITU fue el primer esfuerzo para estandarizar las comunicaciones en varios países.

Años más tarde, en 1884 al otro lado del Atlántico, en Estados Unidos se funda la IEEE (Institute of Electrical and Electronics Engineers), organismo encargado hoy en día de la promulgación de estándares para redes de comunicaciones. En 1906, en Europa se funda la IEC (International Electrotechnical Commission), organismo que define y promulga estándares para ingeniería eléctrica y electrónica.

En 1918 se funda la ANSI (American National Standards Institute), otro organismo de gran importancia en la estandarización estadounidense y mundial. En 1932, al fusionarse dos entidades de la antigua ITU, se crea la Unión Internacional de Telecomunicaciones, otro organismo de gran importancia hoy en día encargado de promulgar y adoptar estándares de telecomunicaciones. Por otra parte, en 1947 pasada la segunda guerra mundial, es fundada la ISO (International Organization for Standardization), entidad que engloba en un ámbito más amplio estándares de varias áreas del conocimiento. Actualmente existe una gran cantidad de organizaciones y entidades que definen estándares.

¿Qué es un estándar?

Un estándar, tal como lo define la ISO "son acuerdos documentados que contienen especificaciones técnicas u otros criterios precisos para ser usados consistentemente como reglas, guías o definiciones de características para asegurar que los materiales, productos, procesos y servicios cumplan con su propósito". Por lo tanto un estándar de telecomunicaciones "es un conjunto de normas y recomendaciones técnicas que regulan la transmisión en los sistemas de comunicaciones". Queda bien claro que los estándares deberán estar documentados, es decir escritos en papel, con objeto que sean difundidos y captados de igual manera por las entidades o personas que los vayan a utilizar.

Tipos de estándares

Existen tres tipos de estándares: *de facto*, *de jure* y los *propietarios*. Los estándares de facto son aquellos que tienen una alta penetración y aceptación en el mercado, pero aún no son oficiales. Un estándar de jure u oficial, en cambio, es definido por grupos u organizaciones oficiales tales como la ITU, ISO, ANSI, entre otras. La principal diferencia en cómo se generan los estándares de jure y de facto, es que los estándares de jure son promulgados por grupos de gente de diferentes áreas del conocimiento que contribuyen con ideas, recursos y otros elementos para ayudar en el desarrollo y definición de un estándar específico. En cambio los estándares de facto son promulgados por comités *guiados* de una entidad o compañía que quiere sacar al mercado un producto o servicio; si tiene éxito es muy probable que una Organización Oficial lo adopte y se convierta en un estándar de jure.

Por otra parte, también existen los "estándares" propietarios que son propiedad absoluta de una corporación u entidad y su uso todavía no logra una alta penetración en el mercado. Cabe aclarar que existen muchas compañías que trabajan con este esquema sólo para ganar clientes y de alguna manera "atarlos" a los productos que fabrica. Si un estándar propietario tiene éxito, al lograr más penetración en el mercado, puede convertirse en un estándar de facto e inclusive convertirse en un estándar de jure al ser adoptado por un organismo oficial.

Tipos de organizaciones de estándares

Básicamente, existen dos tipos de organizaciones que definen estándares: Las organizaciones oficiales y los consorcios de fabricantes. El primer tipo de organismo está integrado por consultores independientes, integrantes de departamentos o secretarías de estado de diferentes países u otros individuos.

Ejemplos de este tipo de organizaciones son la ITU, ISO, ANSI, IEEE, IETF, IEC, entre otras. Los consorcios de fabricantes están integrados por compañías fabricantes de equipo de comunicaciones o desarrolladores de software que conjuntamente definen estándares para que sus productos entren al mercado de las telecomunicaciones y redes (ATM Forum, Frame Relay Forum, Gigabit Ethernet Alliance, ADSL Forum, etc.). Una ventaja de los consorcios es que pueden llevar más rápidamente los beneficios de los estándares promulgados al usuario final, mientras que las organizaciones oficiales tardan más tiempo en liberarlos. Otro aspecto muy importante de los consorcios de fabricantes es que éstos tienen un contacto más cercano con el mundo real (y productos reales). Esto reduce el riesgo de crear especificaciones que son demasiado ambiciosas, complicadas, y costosas de implementar.

¿Cuándo es Oficial un Organismo?

En Estados Unidos, donde se aglutinan la mayoría de las organizaciones, la mejor manera para saber si una organización de estándares es oficial consiste en conocer si la organización está avalada por la ISO; ANSI, IEEE e IETF, todas ellas están reconocidas por la ISO y por lo tanto son organismos oficiales. En el resto del mundo, aquellas organizaciones avaladas por la ITU o ISO son organizaciones oficiales. A continuación se describirán brevemente algunas de las organizaciones de estándares más importantes.

La Unión Internacional de Telecomunicaciones

La ITU es el organismo oficial más importante en materia de estándares en telecomunicaciones y está integrado por tres sectores o comités: el primero de ellos es la ITU-T (antes conocido como CCITT, Comité Consultivo Internacional de Telegrafía y Telefonía), cuya función principal es desarrollar bosquejos técnicos y estándares para telefonía, telegrafía, interfaces, redes y otros aspectos de las telecomunicaciones. El segundo comité es la ITU-R (antes conocido como CCIR, Comité Consultivo Internacional de Radiocomunicaciones), encargado de la promulgación de estándares de comunicaciones que utilizan el espectro electromagnético, como la radio, televisión UHF/VHF, comunicaciones por satélite, microondas, etc. El tercer comité ITU-D, es el sector de desarrollo, encargado de la organización, coordinación técnica y actividades de asistencia.

La IEEE

Fundada en 1884, la IEEE es una sociedad establecida en los Estados Unidos que desarrolla estándares para las industrias eléctricas y electrónicas, particularmente en el área de redes de datos. Los profesionales de redes están particularmente interesados en el trabajo de los comités 802 de la IEEE. El comité 802 (80 porque fue fundado en el año de 1980 y 2 porque fue en el mes de febrero) enfoca sus esfuerzos en desarrollar protocolos estándares para la interfase física de las conexiones de las redes locales de datos, las cuales funcionan en la capa física y enlace de datos del modelo de referencia OSI. Estas especificaciones definen la manera en que se establecen las conexiones de datos entre los dispositivos de red, su control y terminación, así como las conexiones físicas como cableado y conectores.

La Organización Internacional de Estándares

La ISO es una organización no gubernamental establecida en 1947, tiene representantes de organizaciones importantes de estándares alrededor del mundo y actualmente conglomerada a más de 100 países. La misión de la ISO es "promover el desarrollo de la estandarización y actividades relacionadas con el propósito de facilitar el intercambio internacional de bienes y servicios y para desarrollar la cooperación en la esfera de la actividad intelectual, científica, tecnológica y económica". Los resultados del trabajo de la ISO son acuerdos publicados como estándares internacionales. Tanto la ISO como la ITU tienen su sede en Suiza.

Conclusión

Día con día las organizaciones oficiales y los consorcios de fabricantes están gestando estándares con el fin de optimizar las actividades de la vida diaria. En la industria global de redes, los fabricantes que puedan adoptar los estándares en sus tecnologías serán los que predominen en el mercado. Los fabricantes tienen dos grandes razones para invertir en estándares. Primero, los estándares crean un nicho de mercado; segundo, los fabricantes que puedan estandarizar sus propias tecnologías podrán entrar más rápido a la competencia. Antes de comprar algún equipo de telecomunicaciones y redes, acuérdesese de los estándares y elija aquellos que han sido adoptados en su país.

La estandarización de los Sistemas de Información

De la misma forma en que existe el problema de comunicación entre 2 sistemas en telecomunicaciones, existe el problema de comunicación entre 2 aplicaciones de software. Los conceptos de sistemas distribuidos, reutilización de códigos, tecnologías escalables y flexibles, sistemas independientes de plataforma, llaman enormemente la atención de las empresas por una razón: disminución de los costos. Desde la década de los 90 y hasta la fecha, diversas compañías dedicadas al desarrollo de sistemas en sus diversas áreas (hardware, software, comunicaciones, lenguajes de programación, protocolos, etc.) se han dado la tarea de desarrollar estándares en el área de los sistemas de información. Más adelante, se presentan 2 arquitecturas de plataformas de desarrollo e implementación de sistemas de información basados en estándares: SunONE y MS.NET, propuestas por Sun Microsystems y Microsoft Corporation, respectivamente.

3.1.2. Arquitecturas de Sistemas de Información con Servicios Web

Los departamentos encargados de manejar los sistemas de información en las empresas en la actualidad, se enfrentan diariamente al problema de administrar y controlar aplicaciones altamente dependientes y monolíticas corriendo detrás de un firewall. Hoy, se topan de frente ante una nueva realidad en el mundo del software: la necesidad de integrar los procesos de sus negocios, traspasando límites corporativos y geográficos que no pueden ser completamente conocidos o anticipados.

La solución que satisface todos estos requerimientos, tiende a converger a una arquitectura de componentes completamente modulares, que proporcionen un servicio específico, el cual pueda ser ligado dinámicamente a otros componentes que hasta ese momento ignoraban mutuamente inclusive su existencia. Otra tendencia (hasta cierto punto natural) es el hacer estos servicios de información a través de Internet (la red más grande y compleja del mundo, que está extendiendo sus alcances hasta niveles de comunicación personal en dispositivos móviles). Esto ha dado como resultado un estándar en el mercado, en donde dichos componentes son conocidos como *Servicios Web* basados en *XML*.

¿Qué son los Servicios Web?

Los Servicios Web son funciones autónomas de negocio que operan sobre el Internet. Están escritos siguiendo especificaciones estrictas para trabajar conjuntamente con otro tipo similar de componentes. Algunas de las funciones mejor establecidas en este momento son mensajería, directorios de las capacidades del negocio, y descripciones de servicios técnicos. Pero otras funciones están ya en desarrollo. Los Servicios Web son importantes para el negocio por que permiten a sistemas en diversas compañías el interactuar uno con el otro, más fácilmente que nunca.

Con los negocios necesitando una colaboración más estrecha entre los proveedores y los clientes, acoplándose con más empresas en un riesgo compartido y alianzas comerciales a corto plazo, persiguiendo oportunidades en nuevas líneas de negocios, y haciendo frente a la perspectiva de más fusiones y adquisiciones, las compañías necesitan la capacidad de ligar sus sistemas rápidamente con los sistemas de otras compañías. Así, los Servicios Web brindan la capacidad de hacer más negocios electrónicamente, con más socios de negocios potenciales, en más y diversas formas que antes, y a un costo razonable.

Debido a que los Servicios Web están escritos de acuerdo a estándares, todas las partes involucradas trabajan a partir del mismo diseño básico. Las compañías entonces agregan valor y ventajas del negocio a ese diseño básico para satisfacer las necesidades de sus clientes. Por ejemplo, una compañía puede ofrecer a sus proveedores la capacidad de monitorear los niveles del inventario de productos que ellos mismos proporcionan de manera que puedan resurtir producto sin que el cliente haga órdenes de compra separadas.

Los Servicios Web proporcionan las funciones básicas de mensajería y descripción de servicio para esta clase de relación electrónica, pero los proveedores podrían ampliar estas características básicas para proporcionar mejores servicios al cliente e inclusive extenderlas a otros socios de negocios, puesto que están contruidos sobre estándares.

También, debido a que los Servicios Web están contruidos con estándares, se hace posible que los desarrolladores de sistemas se incorporen rápida y fácilmente al mercado, lo que incrementa la competitividad y reduce los costos. La competencia entre vendedores también motiva la innovación en los productos y servicios ofrecidos a los clientes del negocio. Además, al basar los sistemas en estándares se ayuda a prevenir la dependencia hacia un vendedor en específico y cierto tipo de computadoras o de software.

Los Servicios Web son aún un trabajo en proceso. Algunos de los estándares son aún nuevos y no están probados exhaustivamente, y muchos de los usos de negocios potenciales están todavía iniciándose. Pero las compañías debieran estar planeando ya sus Servicios Web y preguntando a sus vendedores cuales son sus planes para soportarlos.

¿Por qué Servicios Web?

Desde luego, una nueva tecnología no tiene sentido si no trae consigo nuevos beneficios que superen a los beneficios que proporcionan las tecnologías ya existentes. Los beneficios claves de los Servicios Web son:

Software como un Servicio

Opuesto a los productos empaquetados, los Servicios Web pueden ser distribuidos y cobrados como un flujo continuo de servicios con acceso ubicuo desde cualquier plataforma. Los Servicios Web permiten el encapsulamiento, los componentes pueden ser aislados de tal manera que sólo los servicios a nivel del negocio son expuestos. Todo esto resulta en poder distribuir componentes y sistemas más estables y más flexibles.

Interoperabilidad Dinámica de Negocios

Nuevas sociedades de negocios pueden ser construidas dinámica y automáticamente gracias a que los Servicios Web pueden asegurar una completa interoperabilidad entre sistemas.

Accesibilidad

Los servicios de los negocios pueden estar completamente descentralizados y distribuidos en Internet, y accesados por una amplia variedad de dispositivos de comunicación.

Eficiencias

Los negocios pueden ser liberados de la enorme carga que representa el complejo, caro y lento desarrollo del software, y en lugar de eso, enfocarse en el valor agregado y las tareas de misión crítica. Los Servicios Web contruidos a partir de aplicaciones hechas para el uso interno pueden ser fácilmente expuestos para uso externo sin cambiar el código. El desarrollo incremental empleando Servicios Web es natural y fácil, y ya que los Servicios Web son declarados e implementados en un formato de lectura para humanos es fácil de depurar y mantener. El resultado total es un riesgo reducido y mayor eficiencia en su puesta en marcha.

Especificaciones Universalmente Acordadas.

Los Servicios Web están basados en especificaciones que han sido acordadas universalmente para un intercambio estructurado de información, mensajería, descubrimiento de servicios, descripción de la interfaz y orquestación de procesos del negocio.

Integración del Legado.

Una mayor agilidad y flexibilidad a partir de una mayor integración entre sistemas ya existentes.

Nuevas Oportunidades de Mercado.

Sin duda habrá mayor viabilidad de negocios a las empresas que sean dinámicas y se crearán nuevas formas de hacer sociedades.

¿Quién se beneficiará del fenómeno de los Servicios Web?

La primera oleada de Servicios Web deberá integrar aplicaciones existentes, primero dentro de la empresa y más tarde con socios externos. Con todos los grandes vendedores moviéndose hacia esta arquitectura, la integración de diferentes soluciones de software será más fácil en su aspecto técnico. Por lo tanto, una solución que sea de lo mejor en su clase, que es a menudo experimentada como un ejercicio técnico de integración que requiere mucho tiempo, se hará más popular. El riesgo aquí está en que los vendedores querrán impedir a sus productos el ser fáciles para integrarse, entonces los clientes comprarán sólo parte de su suite de producto. Esto es sin embargo, sólo un pequeño riesgo, por que los clientes exigirán franqueza con Servicios Web de parte de sus vendedores.

Y desde luego, el habilitar Servicios Web será un importante punto de venta. Otro riesgo es el futuro de los estándares. Aunque los estándares básicos están ya aceptados, los vendedores desarrollan nuevos estándares por su cuenta, lo cual resultará en riesgos de incompatibilidad.

Los productos bien posicionados, así como sus vendedores, prosperarán en esta arquitectura de componentes distribuidos, ya que la demanda de productos especializados crecerá. Desde luego, hay más motivos que la conectividad para tomar una decisión sobre un acercamiento de un único vendedor o comprar de múltiples vendedores. Sin embargo la cuestión de la conectividad es una de las razones más importantes para impedir que los compradores usen la mejor solución de su clase. Los Servicios Web definitivamente ayudarán a cambiar esto. La ventaja de tener más de un vendedor es la reducción en la dependencia. Algo importante a tener en mente es que esto no es siempre una razón para elegir múltiples vendedores. En muchos casos donde las compañías se fusionan y múltiples sistemas necesitan ser integrados, los Servicios Web desempeñarán un papel muy importante.

Los problemas de integración han sido siempre cuestiones muy técnicas, aunque la parte más importante es la parte comercial y de negocios de la integración.

En el futuro, los integradores de sistemas se concentrarán más en la arquitectura para la integración de la aplicación y los vendedores se concentrarán más en la parte técnica de poner los Servicios Web a disposición de ese integrador. El crecimiento de los Servicios Web podría resultar también en la capacidad de las compañías para integrar sistemas por sí mismos, ya que se requieren conocer menos detalles específicos del mismo.

La integración será definitivamente más fácil. La pregunta es, ¿quien va a ganar mayores beneficios?: ¿el cliente al tener sistemas mejor integrados?; ¿los vendedores mejor posicionados vendiendo más software?; ¿o el integrador de sistemas haciendo más integraciones que sean lo mejor de su clase?. La respuesta es: más integración con más vendedores y con una más alta calidad. Todos serán ganadores.

Los Componentes de los Servicios Web

Los Servicios Web y XML

A pesar de su sencillez aparente, XML está transformando completamente la creación y el uso de software. El Web revolucionó la comunicación entre usuarios y aplicaciones. XML está revolucionando la comunicación entre aplicaciones o, de forma más general, la comunicación entre equipos, pues ofrece un formato de datos universal que permite adaptar o transformar fácilmente la información.

Los servicios Web XML permiten que las aplicaciones compartan información. Los servicios Web XML son unidades de código, cada una de las cuales se encarga de un conjunto limitado de tareas, están basados en XML, el lenguaje universal del intercambio de información en Internet y pueden utilizarse en cualquier plataforma o sistema operativo.

XML: Extensible Markup Language

XML agrega un cierto grado de inteligencia a los componentes Web ya que describe el contenido de la información mediante el uso de meta-datos o etiquetas. Esto permite a los servicios Web comunicarse entre sí al intercambiar mensajes en forma de documentos XML. XML usa texto plano para la representación de datos, haciendo sencilla su lectura tanto para humanos como para máquinas de búsqueda. Cualquier servicio que pueda procesar un archivo XML y devolver mensajes en formato XML es capaz de comunicarse con cualquier otro servicio o aplicación que pueda hacer lo mismo. Este tipo de arquitectura permite nuevos niveles de interoperabilidad y una cantidad virtualmente ilimitada de opciones para la creación de servicios de valor agregado.

Los servicios Web XML: Un lenguaje universal

Los servicios Web XML permiten que las aplicaciones compartan información y que además invoquen funciones de otras aplicaciones independientemente de cómo hayan sido creadas, cuál sea el sistema operativo o la plataforma en que se ejecutan y cuáles los dispositivos utilizados para obtener acceso a ellas. Aunque los servicios Web XML son independientes entre sí, pueden vincularse y formar un grupo de colaboración para realizar una tarea determinada.

El lenguaje de los Servicios Web

Los servicios Web requieren nuevos niveles de inteligencia que van más allá de HTML y TCP/IP para dar soporte a la transferencia de información entre sistemas y dispositivos no relacionados directamente entre sí. Las reglas que controlan el descubrimiento, la descripción y el intercambio de información están siendo definidas por nuevos estándares como XML (eXtensible Markup Language), ebXML (electronic business XML), UDDI (Universal Discovery, Description and Integration Specification), SOAP (Simple Object Access Protocol) y WSDL (Web Services Description Language). La meta es eliminar las barreras de comunicación entre una multitud de dispositivos y de sistemas que existen actualmente, así como facilitar el envío de servicios Web independientemente del proveedor. Gartner research ha identificado 5 requerimientos iniciales para una plataforma de servicios Web:

1. Descubrimiento: el mecanismo por el cual los servicios se hacen conocidos por sí mismos y pueden ser descubiertos por otros.
2. Descripción: La especificación sobre la forma en que la información entra hacia y sale del servicio.
3. Transporte: El método de comunicación entre el servicio y el usuario final.
4. Entorno: El entorno de ejecución en que el servicio es ejecutado. La plataforma de e-business describe el entorno.
5. Notificación de eventos: Un mecanismo mediante el cual se puede invocar al servicio como parte de una serie de eventos *encolados* o *secuenciales*.

UDDI: Universal Description, Discovery and Integration

Prácticamente todos los sistemas de cómputo distribuido tienen algún servicio de resolución de nombres, que es esencialmente el componente que permite al productor de un servicio publicar la existencia de dicho servicio en un lugar previamente definido y bien conocido. Un consumidor potencial del servicio puede dirigirse a este lugar bien conocido, buscar el servicio y encontrar la forma de conectarse con él. UDDI y el Registro/Repositorio ebXML son 2 diferentes servicios globales de resolución de nombres que pueden actuar como registro y repositorio para servicios Web (Sun Microsystems originalmente se concentró en soportar UDDI, pero posteriormente incluyó el soporte para el registro/repositorio ebXML, ya que cuenta con más prestaciones).

SOAP: Simple Object Access Protocol

SOAP es una tecnología desarrollada por DevelopMentor, IBM, Lotus, Microsoft y Userland. SOAP proporciona un protocolo de mensajes extensible sobre XML y soporte para un modelo de programación RPC (Remote Procedure Call). Existen en la actualidad varias implementaciones disponibles de SOAP, las 2 más populares son: una implementación de código abierto hecha en Java por Apache Software Foundation y la implementación de Microsoft, incluida en el .NET SDK.

WSDL: Web Services Description Language

Esta tecnología, desarrollada por Ariba, IBM y Microsoft, especifica un esquema común en XML para describir un servicio Web. A la fecha, IBM ha liberado un kit de herramientas para desarrolladores de WSDL basado en Java, incluido en IBM alphaWorks.

Seguridad y Servicios Web XML

Un beneficio clave en la nueva arquitectura de servicios Web es la habilidad de entregar soluciones integradas e interoperables. Ayudar a proteger la integridad, confidencialidad y seguridad de los servicios Web a través de la aplicación de un modelo comprensivo de seguridad es crítico.

Ventajas de los servicios Web XML:

- Abren la puerta a nuevas oportunidades empresariales, pues facilitan la comunicación con los socios.
- Ofrecen a los usuarios experiencias mucho más personalizadas e integradas, por medio de la nueva gama de dispositivos inteligentes, como las PC.
- Ahorran tiempo y dinero, ya que reducen la duración del ciclo de creación.
- Aumentan los flujos de ingresos, pues ponen fácilmente sus propios servicios Web XML a disponibilidad de otros.

A continuación se explican con mayor detalle algunos aspectos de seguridad de Servicios Web

3.1.3. Seguridad y Servicios Web

Para que una tecnología que ofrece servicios de información a gran escala (como son los Servicios Web) pueda tener éxito, indudablemente debe ser una tecnología que nos ofrezca seguridad. Para poder hablar de la seguridad en los Servicios Web, primero hay que definir lo que queremos decir por "Seguridad".

Una forma sencilla de definir el concepto de seguridad en el contexto de Servicios Web, es mencionando las necesidades básicas que se buscan satisfacer... pero, ¿cuáles son esas necesidades?

Integridad: La integridad de la información se refiere a la certeza que tiene el receptor de que el contenido del mensaje que recibió es idéntico al que el transmisor envió por el medio de transmisión, es decir, que durante el viaje no haya sufrido modificaciones de ningún tipo.

Confidencialidad: La confidencialidad garantiza que la información que se transmite llega únicamente a aquellos receptores a quienes está destinada, y a nadie más.

Identidad del Transmisor: Muy frecuentemente es requisito indispensable garantizar al 100% que un transmisor es en realidad quien dice ser para poder confiar en la información que este emite.

Autorización: En un entorno de aplicaciones heterogéneas altamente integradas, es indispensable saber qué usuarios están autorizados a realizar qué operaciones dentro del sistema.

En la práctica, en un sistema distribuido es relativamente sencillo satisfacer las 2 primeras condiciones utilizando criptografía como infraestructura base (específicamente firmas digitales y algoritmos de cifrado). Sobre estas tecnologías, se pueden construir soluciones adicionales para satisfacer los últimos 2 requerimientos.

Actualmente existen muchas organizaciones desarrollando estándares de seguridad en servicios Web, pero a continuación se mencionarán los trabajos más importantes:

XML Digital Signature (Firma Digital XML): Es un estándar cuya especificación ya fue terminada, desarrollado como un proyecto de cooperación entre la W3C y la IETF. Este estándar define un método de autenticidad de entidades por firmas digitales basado en XML.

XML Encryption (Cifrado XML): Es un grupo de trabajo de la W3C, que también cuenta con especificaciones terminadas. En ellas, se define la forma de cifrar un documento XML o sólo porciones del mismo.

XML Key Management: Un trabajo de la W3C que define métodos para el manejo de claves y certificados en XML, mismos que permiten manejar aspectos como la identidad, autenticidad, autorización y revocación de permisos, etc. El trabajo está en sus fases finales y las especificaciones preliminares no han tenido problemas.

OASIS Security Services (Servicios de Seguridad OASIS): También conocido como SAML (Security Authorization Markup Language), es un entorno de intercambio de información que permite escenarios de *Single Sign On (SSO)* en sistemas distribuidos; esto es: en un sistema distribuido, un objeto puede tener identidad, autorización, roles y permisos diferentes en cada subsistema (o módulo del sistema) o inclusive en varios sistemas diferentes. SAML permite que el objeto se autentifique una sola vez y las autentificaciones subsiguientes necesarias entre los diferentes subsistemas se hacen de forma automática mediante SAML y de manera transparente para el objeto, el cual tiene la percepción de que tiene una identidad única en la totalidad del sistema o sistemas distribuidos. La especificación ya está terminada.

XML Access Control Markup Language (Control de Acceso XML): XACML es un entorno que permite asignar permisos a los objetos dentro de un sistema distribuido. Este lenguaje soporta condiciones que no implementan fácilmente otras facilidades de control de acceso, pues considera además prioridad con criterios como por ejemplo: por día de la semana, por hora del día, por vacaciones, etc. El proyecto está en fase final y no se esperan cambios significativos en la especificación.

OASIS Digital Signature Services (Servicios de Firma Digital): Es un grupo de trabajo que tiene la tarea de desarrollar un estándar que permita el manejo de firmas digitales de forma distribuida (es decir, que no estén basadas en un repositorio central).

OASIS Web Services Security (Seguridad de Servicios Web): Está concebida en base a un documento sobre seguridad de IBM y Microsoft, el cual define la forma de firmar un mensaje SOAP. Adicionalmente, se trabaja para implementar un esquema de seguridad a más alto nivel que incluya características como la integración de políticas e interoperabilidad automática.

Adicionalmente, "The Web Services Interoperability Organization" (Organización para la Interoperabilidad de los Servicios Web) trabaja en un esquema de seguridad que permita la comunicación entre Servicios Web de diferentes proveedores. En este trabajo, además de WS-Security se han definido 6 nuevas especificaciones (WS-Policy para definir políticas de seguridad, WS-Trust para establecer niveles de confianza entre Servicios Web, WS-Privacy que define mecanismos de protección de la información, WS-SecureConversation que permitirá establecer canales de comunicación seguros entre 2 o más Servicios Web, WS-Federation, que permitirá el manejo de diferentes identidades de objetos a través de sistemas heterogéneos y distribuidos, y finalmente WS-Authorization para manejar los permisos de las operaciones de cada objeto en el sistema)

La buena noticia es, que a pesar de que muchas especificaciones se encuentran en etapas finales, no es necesario esperar a que se liberen, actualmente ya podemos construir soluciones para problemas del mundo real.

¿Y por qué no usar SSL?

Supongamos que queremos encontrar solución al problema de proteger un mensaje SOAP que viajará a través de la red. La pregunta obvia es: ¿por qué SSL no es suficiente? Cuando Netscape envió la propuesta de SSL a la IETF, se llamó al grupo de trabajo "Seguridad de la Capa de Transporte" por una buena razón.

Primero: SSL opera entre un punto de comunicación a otro y no entre una aplicación y otra, lo cual es suficiente para la aplicación más popular de la red, en que un usuario se dirige a un sitio Web para visualizar contenido mediante un navegador. Aún en estos casos el esquema puede ser insuficiente. Imaginemos un script CGI, o una vulnerabilidad del servidor Web que ocasionen que los datos enviados se corrompan sin que ello se pueda ser detectado.

Además, en ocasiones en las que es necesario tener varias instancias de servidores Web en un sistema, la configuración puede ser complicada, engorrosa y propensa a errores.

SSL/TSL no tiene mecanismos que permitan "guardar" un mensaje para corroborar su validez o integridad más tarde.

Cuando se establece una conexión de SSL, ambas partes (transmisor y receptor) manejan una llave por un periodo corto de tiempo. En el caso de un error de comunicación, es imposible para cualquiera de las 2 partes asegurar que tienen en su poder el mensaje correcto.

Por todo ello, SSL resulta inapropiado e insuficiente para asegurar un mensaje SOAP.

3.1.4. Arquitectura SunONE de Sun Microsystems

“Services on Demand” (Servicios en Demanda) es una visión que deriva del reconocimiento de que servicios de muy diversos tipos deberán ser entregados a los usuarios a través de redes de comunicación electrónicas, incluyendo una nueva clase de aplicaciones que van más allá de las fuertemente dependientes y monolíticas que conocemos en la actualidad. Los “Services on Demand” (SOD) harán más fácil la entrega de servicios de cómputo en cualquier lugar, a cualquier persona, sin importar el tipo de dispositivo que use; los SOD evolucionarán rápidamente hasta incluir soporte para las más novedosas aplicaciones de servicios Web y punto a punto. Sun proporciona una plataforma *end-to-end* que hará posible a sus clientes satisfacer las necesidades de sus aplicaciones actuales a la vez de dotarlos de bases sólidas para soportar los servicios Web del futuro. SunONE (Sun Open Network Environment) es una plataforma abierta, integrable y modular diseñada para crear y entregar SOD y los nuevos servicios Web. La mayoría del software base requerido para soportar SOD ya está disponible. La habilidad de entregar servicios Web requiere de un enfoque que difiere enormemente del modelo cliente-servidor que casi cualquier aplicación usa actualmente.

Servicios en Demanda y SunONE, ¿Es esto realmente diferente?

Los protocolos de aplicación (que son las reglas establecidas para el flujo de información a través de Internet) son fundamentalmente diferentes de los servicios Web. El cómputo cliente-servidor fue diseñado en base a un modelo de Intranet, donde un servidor que está en el tercer piso de un edificio puede dar servicio a 500 estaciones de trabajo que se encuentran en el segundo piso. El desarrollo del software en este entorno estaba basado en componentes que corrían detrás de un firewall, y que usaban tecnologías como DCOM, CORBA, J2EE y EJB. Este modelo no fue pensado para, por ejemplo, permitir a un PDA Palm comunicarse con un servicio de mapas por Internet como MapQuest.

La primera generación de servicios en Internet típicamente ha usado protocolos como TCP/IP en conjunto con otros protocolos de aplicación poco o nada flexibles. Los servicios Web, en cambio, funcionan con un modelo de cómputo distribuido que es tanto flexible como extensible. Estos servicios utilizan principalmente HTTP como mecanismo de transporte y XML como protocolo de aplicación. Una de las mayores diferencias entre los servicios Web y las aplicaciones cliente-servidor está en los protocolos que se usan en cada uno de ellos. Los servicios Web se construyen sobre, y en coexistencia con sistemas de cómputo distribuido tradicionales.

Componentes vs. Servicios Web

Los servicios Web proporcionan un modelo abstracto que describe un mecanismo para que el productor de algún servicio pueda hacerle llegar información a un cliente que la quiera usar. Este código tiene que escribirse para implementar dicho servicio, inclusive puede ser que los programadores de un lenguaje orientado a objetos vean esto como otro modelo más de componentes, sin embargo los servicios Web se diferencian en dos aspectos muy importantes: la *granularidad* y el *acoplamiento*.

La *granularidad* se refiere a la complejidad de la descripción de un servicio. Por ejemplo: una llamada de sistema simple en UNIX puede ser muy específica, teniendo con esto un nivel muy fino de "granularidad". Una orden de compra de un fabricante a un proveedor, en cambio, puede tener implicaciones globales que introducen niveles de complejidad que no están completamente bien definidos, resultando en un nivel de "granularidad" más grueso.

El acoplamiento se refiere a la naturaleza de las interfaces de comunicación entre el productor y el consumidor del servicio. Tiene mucho que ver con el impacto que se tendría en el consumidor si la implementación de la aplicación llegara a cambiar. Por ejemplo, si Sun decidiera cambiar la llamada de sistema de Solaris "GetTimeOfDay", cualquier aplicación que la usara tendría que adaptarse a los cambios. Este es un ejemplo de aplicaciones fuertemente ligadas y dependientes.

Por otro lado, los componentes sirven al propósito de crear aplicaciones con una "granularidad" fina y fuertemente dependientes. Los servicios Web en cambio, sirven para crear aplicaciones con "granularidad" gruesa y débilmente dependientes. Otra manera de verlo, es que los componentes (fuertemente dependientes) son capaces de interactuar con otros componentes siempre y cuando se usen las mismas API's o modelos de objetos, mientras que los servicios Web (débilmente dependientes) están diseñados para interactuar con cualesquiera otros componentes sin importar su origen, siempre y cuando estén encapsulados en una *envoltura* (o wrapper) auto descriptiva en XML.

Servicios en Demanda y SunONE 5

Para los programadores, esta plataforma crea ciertos límites en el diseño. La principal razón de que esto ocurra se debe a las características del mecanismo de transporte: TCP/IP o HTTP. Entonces, la respuesta a la pregunta de un programador sobre si requiere un componente o un servicio Web en realidad depende de la naturaleza de la petición al servicio que se necesita. Los servicios Web no reemplazarán completamente a los componentes de software, sino que permitirán extenderlos y complementarlos. La siguiente tabla compara las características principales de cada uno de estos modelos de programación:

Modelo basado en componentes	Modelo basado en servicios Web
Aplicaciones fuertemente acopladas	Aplicaciones débilmente acopladas
Software basado en Intranet	Software basado en Internet
Diseñado para procesos dentro de la empresa	Diseñado para procesos entre empresas
Mecanismo de transporte: TCP/IP	Mecanismo de transporte: HTTP
"Granularidad" fina	"Granularidad" gruesa

Manejadores y Tendencias del Mercado

El Mercado de los Dispositivos

Los servicios en demanda están siendo determinados por ciertas tendencias. La primer tendencia más importante es debido a la amplia adopción de dispositivos móviles, como teléfonos celulares y PDA's, que requieren de nuevos servicios de software que puedan ser utilizados a través de Internet.

El Estancamiento del Mercado de las PC's

La segunda tendencia es el declive de las PC's y el software con licencias, que ha hecho que muchas compañías, como Microsoft, abandonen este esquema de licenciamiento, buscando una alternativa mediante la suscripción de servicios.

Dispositivos de Escritorio vs. Dispositivos Móviles

En la actualidad, no sólo requerimos aplicaciones portátiles, además nos comunicamos como nunca antes, en especial las personas que frecuentemente están lejos de sus escritorios y necesitan usar servicios de mensajería de texto y de voz. A diferencia del software para equipos de escritorio que asume cosas sobre el usuario (como por ejemplo: que conoce su ubicación física exacta y el perfil del sistema que usará), los dispositivos portátiles como celulares y PDA's no tienen una configuración que se pueda predecir: requieren contar con la habilidad de modificar sus interfaces con el usuario de forma dinámica, tener funcionalidad modular así como poder identificarse automáticamente con otros componentes y dispositivos.

Personalización: Una Mejor Experiencia del Usuario

Los usuarios gozarán de beneficios adicionales en términos de riqueza de la experiencia con el sistema, al poder personalizar los servicios que use. Los servicios en demanda tienen el potencial de conocer el contexto del usuario, es decir: su perfil y sus preferencias de acuerdo al lugar y hora en que se encuentre. La personalización también provee un cierto nivel de inteligencia a un sistema, dándole la posibilidad de combinar componentes y servicios para entregar al usuario exactamente la información en que está interesado.

Bajos Costos y Alta Flexibilidad

La tendencia en el mercado del software, se dirige a la compra de servicios en lugar del desarrollo de aplicaciones, esto por el bajo costo y la alta flexibilidad que permite.

El Modelo de Negocios para los Servicios Web

Los servicios en demanda, están encaminados a satisfacer las necesidades de dos muy grandes mercados: Negocio-Consumidor (B2C por sus siglas en inglés "Business-to-Consumer") y Negocio-Negocio (B2B por "Business-to-Business"). Mientras que la tecnología base y el mecanismo de entrega de los servicios es la misma, el usuario final y los beneficios resultantes variarán. Existirá una importante oportunidad de negocios para compañías que sean capaces de intercambiar servicios en Internet con cualquiera de estos modelos.

Servicios Web B2C

Mientras se consoliden los servicios Web, los consumidores irán teniendo oportunidad de interactuar con mucho más que con una tienda en línea basada en Web. Dentro de poco, un simple conjunto de opciones personalizadas facilitarán el intercambio de una gran cantidad de transacciones en cascada que operarán automáticamente en beneficio del consumidor.

Basado en un perfil de opciones individual, la compra de un boleto de avión podría desencadenar automáticamente la reservación de una habitación de hotel y renta de automóvil, mismas que podrán enviar una confirmación de reservación por correo electrónico al consumidor. Mayores oportunidades existen en el área del entretenimiento, donde los entusiastas de los juegos tendrán la oportunidad de interactuar con otros jugadores en línea en una gran variedad de entornos por computadora.

Servicios Web B2B

Actualmente, muchas firmas consultoras están de acuerdo en que las aplicaciones de negocios viables para los servicios Web llegarán antes que las aplicaciones para consumidores. En un informe, la consultora H&Q dice que en un ambiente de negocios cuya transformación está determinada por Internet, las compañías encontrarán maneras de encapsular las aplicaciones base para su posición en el mercado (préstamos a empleados, distribución de las ventas, entrega de paquetes, etc.) en servicios Web accesibles. Estos servicios tendrán la capacidad de combinarse y crear una red más compleja de servicios basados en Web.

XML + Java = Servicios Web J2EE

XML y Java son 2 tecnologías clave que permiten la arquitectura de servicios Web débilmente ligados. Juntos, proporcionan un entorno de desarrollo y entrega de servicios Web estandarizado e independiente de plataforma. Java proporciona un entorno de ejecución de software portable, independiente de plataforma, mientras que XML permite una representación de datos portable e independiente de plataforma. Las plataformas J2ME, J2SE y J2EE integran tecnología XML como parte fundamental, y esto garantiza a proveedores y consumidores la libertad de elección de sus productos, servicios y componentes, eliminando también el riesgo de tener que "casarse" con un sólo proveedor (como ocurre con MS.NET). Java 2 y XML en conjunto, permiten a los clientes "ensamblar" su propio paquete de soluciones, servicios y software basado en estándares, eligiendo lo mejor de lo mejor entre los diversos proveedores.

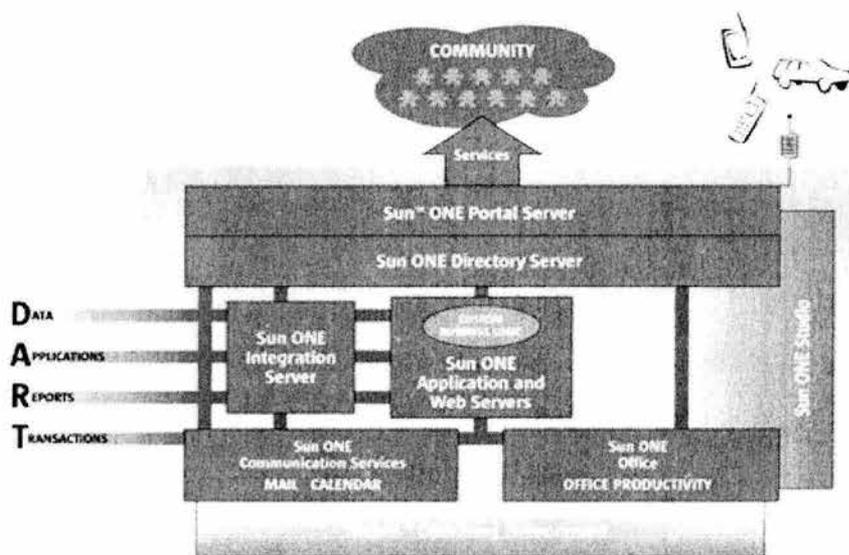
¿Qué es SunONE?

SunONE es la plataforma de software recomendada por Sun para entregar servicios en demanda. SunONE (Sun Open Network Environment) es la visión, la plataforma, los productos y el conocimiento de Sun para construir y entregar servicios en demanda. SunONE representa una solución de software comercial para implementar una infraestructura de desarrollo de servicios en demanda. Los servicios en demanda están basados en estándares abiertos que pueden ser usados por socios, consumidores y proveedores sin ningún cambio en la calidad y capacidades de cada servicio. SunONE es una pila de productos abiertos e integrables para crear y entregar servicios en demanda: desde el entorno operativo Solaris y las herramientas Forte, pasando por Java como plataforma de desarrollo, y llegando al conjunto de productos y soluciones iPlanet (iPlanet Application Server, iPlanet Portal Server, iPlanet Integration Server, iPlanet Directory Server y aplicaciones de comercio de iPlanet). La especialización en SunONE incluye servicios profesionales de Sun, el diseño de una arquitectura para facilitar la entrega de servicios, iForce como la estrategia de soluciones al consumidor de Sun, así como SunTone que provee una base óptima para el desarrollo de servicios en demanda.

Análisis

El diagrama "DART" de SunONE

Si usted se pregunta, ¿de dónde vienen estos servicios de los que hemos hablado? La respuesta es que están contruidos por fuera de las fuentes de información. Sun usa el acrónimo DART para los Datos, Aplicaciones, Reportes y Transacciones (Data, Applications, Reports, Transactions) y representan a las fuentes de información, así como la arquitectura que puede dar vida a esas fuentes en forma de servicios. En pocas palabras, DART es una lista de cosas que nos permiten empezar a pensar sobre la forma en cómo estas fuentes están disponibles, y cómo pueden ser potenciadas por una plataforma de software para satisfacer necesidades, nuevas o ya existentes, en el área de los negocios. Esta arquitectura comienza arriba, centrándose en la comunidad: ¿a quién necesita comunicar información un negocio?, ¿a sus empleados en todo el mundo? ¿a sus socios locales?. Los negocios orientan sus actividades hacia esas comunidades primero que nada. Entonces, para contactar a esas comunidades, la arquitectura ha incluido un elemento que puede presentar y agregar datos cohesiva y atractivamente; esa es la parte de portal en la arquitectura, y es una de las formas en que las comunidades disponen de los datos a través de Internet. Otra forma de administrar esos datos, es a través de un directorio capaz de almacenar toda la información de un usuario, como por ejemplo, quien es, qué privilegios o permisos tiene, y con qué componentes del negocio puede interactuar.



El diagrama DART de SunONE

3.1.5. Arquitectura Microsoft .NET

Definiendo los elementos básicos de .NET

Microsoft .NET es un conjunto de tecnologías de software de Microsoft para conectar su mundo de información, gente, sistemas y dispositivos. Permite un nivel sin precedente de integración de software a través del uso de servicios Web XML: pequeños y discretos bloques de aplicaciones que se conectan entre sí y a otras aplicaciones grandes vía Internet.



Esquema de conexión de la plataforma Microsoft .NET

Los componentes del software conectado de Microsoft .NET

.NET está conformada por varios productos de Microsoft que proporcionan la habilidad para construir, hospedar e implementar de una manera rápida y confiable servicios Web, utilizando soluciones seguras e interconectadas. La plataforma Microsoft provee una suite de herramientas de desarrollo, aplicaciones cliente, servicios Web XML, y servidores necesarios para intervenir en este mundo conectado.



El software de aplicaciones para clientes "inteligentes" (móviles) y sistemas operativos permiten a las PC's y otros dispositivos inteligentes interactuar con los servicios Web XML, teniendo acceso a la información en cualquier lugar y en cualquier momento.



Microsoft en colaboración con otras compañías, están desarrollando un conjunto básico de servicios Web XML que van desde servicios de autenticación hasta servicios de agenda y que pueden ser combinados con otros servicios Web XML o usados directamente con aplicaciones del cliente inteligente.

Análisis

Microsoft MapPoint .NET, un servicio Web XML que le permite integrar mapas de alta calidad, indicaciones viales, y otras locaciones inteligentes en sus aplicaciones, procesos de negocio, y sitios Web, es un ejemplo de uno de estos servicios.

Servidores



Microsoft provee una completa infraestructura en servidores (incluyendo la familia de servidores Microsoft Windows 2000 y los Servidores .NET Enterprise) para implementar, manejar e instrumentar los servicios Web XML.

Herramientas de desarrollo



Microsoft Visual Studio .NET y el Microsoft .NET Framework son la solución completa para que los desarrolladores puedan construir, implementar y ejecutar los servicios Web XML.

Software de cliente inteligente y .NET

Tanto si utilizan un equipo de escritorio como un dispositivo inalámbrico en el exterior, los usuarios deberían obtener acceso a la información necesaria y utilizarla cuando la necesiten. Esta experiencia informática libre de problemas se está haciendo realidad con el uso, por parte de los programadores, de la plataforma Microsoft .NET para crear aplicaciones con una gran variedad de características que se integran con Windows e Internet. Microsoft .NET ofrece esta experiencia por medio del software de cliente inteligente.

Una nueva generación de aplicaciones conectadas (clientes inteligentes) posibilitará el funcionamiento de equipos y dispositivos móviles e incrustados y permitirá una experiencia de usuario más personalizada en la plataforma .NET. Con Microsoft Windows XP, Windows XP Embedded y Windows Mobile .NET, Microsoft está creando esta nueva generación de clientes de software que permiten la conexión en cualquier lugar y en cualquier momento.

El entorno de ejecución seguro y administrado de .NET Framework permite a los programadores crear aplicaciones para el software de cliente en una gran variedad de dispositivos inteligentes.

Dispositivos inteligentes y .NET

Entre los dispositivos inteligentes figuran los PC's, equipos portátiles, estaciones de trabajo, teléfonos, PDA's, Tablet PC's, consolas de juegos Microsoft Xbox y equipos que por ahora sólo imaginamos. Estos dispositivos son "inteligentes" porque son capaces de obtener acceso a servicios Web XML y le permiten interactuar con sus datos independientemente de la ubicación, el tipo y el número de dispositivos que utilice. Los dispositivos inteligentes le permiten obtener acceso a su información en el formato adecuado en cualquier momento y lugar utilizando los servicios Web XML y ofrecen experiencias en .NET. Todos los dispositivos inteligentes optimizan la presentación y recopilación de la información, desde la conversión de texto a voz, al reconocimiento de la escritura manual.

Los Servidores de Microsoft y .NET

Con Microsoft .NET, los negocios pueden convertir los procesos clave, en servicios Web XML. Ya sea el itinerario de la oficina de un doctor, el catálogo para un vendedor de libros o una calculadora de intercambio de promedios de divisas a nivel mundial, los servicios Web XML creados y alojados por una compañía o individuo pueden ser usados por millones de personas en varias combinaciones, para generar experiencias computacionales altamente personales e inteligentes. Este modelo distribuido de cómputo incrementará las demandas en las infraestructuras de servidores. Servidores seguros y escalables, que integren XML, proporcionarán la base para alojar e implementar software y servicios .NET. Microsoft Windows 2000 Server, la nueva familia de Windows Server 2003, y Microsoft .NET Enterprise Servers son los mejores servidores que Microsoft puede ofrecer para cumplir con esta función.

Infraestructura de Servidor

Microsoft .NET Enterprise Servers, la familia de Windows 2000 Server y la nueva familia de Windows Server 2003, proporcionan la mejor solución para alojar e implementar servicios Web XML, con su seguridad integrada, su soporte de XML y su capacidad para escalar rápidamente y enfrentar demandas en aumento.

Las Herramientas del Desarrollador y .NET

Con Microsoft Visual Studio .NET y .NET Framework, los desarrolladores pueden rápidamente crear servicios Web XML e integrarlos fácilmente a otras aplicaciones. La mayoría de los desarrolladores pueden aprovechar habilidades existentes, debido a que el entorno de ejecución del lenguaje común del .NET Framework le permite desarrollar servicios Web usando cualquier lenguaje de programación moderno.

Microsoft Visual Studio .NET y Microsoft .NET Framework proveen una solución completa para que los desarrolladores puedan construir, implementar y ejecutar servicios Web XML. Estas herramientas incrementan al máximo el rendimiento, la confiabilidad y la seguridad de los servicios Web XML.

Reuso de Servicios Web XML

El desarrollo con servicios Web XML es similar al desarrollo con componentes. Visual Studio .NET le da a los desarrolladores la facilidad de importar servicios Web XML o usar servicios Web XML alojados remotamente y programarlos como lo harían con un elemento COM hoy en día, ahorrando tiempo y dándole a los desarrolladores la oportunidad de concentrarse en funcionalidades muy específicas.

Microsoft .NET Framework y Microsoft .NET Compact Framework

.NET Framework y el dispositivo enfocado a .NET Compact Framework, son ambientes de ejecución de aplicaciones de alta productividad, basados en estándares y multi-lenguajes que manejan tareas esenciales y facilitan el desarrollo. El ambiente de ejecución de la aplicación administra la memoria, trata problemas de versiones y mejora la confiabilidad, escalabilidad y seguridad de sus aplicaciones. Los componentes incluyen un ambiente de ejecución de lenguaje común, un grupo de librerías de clases para desarrollar servicios Web XML y ASP.NET.

Análisis

El ambiente de ejecución de lenguaje común, es el motor en .NET Framework que proporciona un ambiente administrado y seguro, y es diseñado para dar soporte a los desarrolladores que usan muchos lenguajes de programación al crear sus aplicaciones. Tiene un sistema unificado y habilita la herencia de lenguaje cruzado y la depuración. Al usar .NET Framework, los desarrolladores tienen un medio rápido y productivo para desarrollar aplicaciones que realmente son servicios y aplicaciones Web XML de tercera generación.

3.2. Justificación: J2EE (SunONE) vs. Microsoft .NET

A continuación se realizará una comparación entre dos de las muchas elecciones que los negocios tienen para construir Servicios Web basados en XML, que si bien no son las únicas, sí son las más importantes y representativas de las tendencias del mercado: SunONE y MS.NET.

El objetivo de esta comparación es tener un conocimiento sólido de estas dos tecnologías y así seleccionar de manera inteligente la que más nos conviene en la elaboración de Servicios Web desde el punto de vista económico, tecnológico y práctico.

Introducción

Según Graham Glass autor de *"La Revolución de los Servicios Web"* éstos se pueden definir como: "Una colección de funciones que están empaquetadas como una única entidad y publicadas en la red para ser usados por otros programas."

Los Servicios Web son bloques de construcción para construir sistemas distribuidos abiertos, y que permiten a las compañías e individuos el poner sus activos digitales disponibles a todo el mundo de manera rápida y barata".

Previo al advenimiento de los Servicios Web la integración de las aplicaciones era muy difícil debido a diferencias en los lenguajes de programación y el "middleware" usado dentro de las organizaciones. La interoperabilidad era incómoda y dolorosa.

Con los Servicios Web cualquier aplicación puede ser integrada mientras esté en Internet. El fundamento de los Servicios Web está en usar mensajes XML sobre protocolos Web estándar como HTTP.

Con los Servicios Web, se pueden integrar dos negocios, departamentos, o aplicaciones rápidamente y a bajo costo.

La visión de los Servicios Web está en crear servicios que se registrarán a sí mismos de manera pública o privada en los registros de los negocios y de la misma forma se describirán completamente por sí mismos (incluyendo la estructura de su interfase, requerimientos y procesos de negocios, así como los términos y condiciones de uso). Los consumidores de estos servicios sólo tendrán que leer esta descripción para entender las habilidades de estos Servicios Web. Para que estos servicios puedan interactuar dinámicamente, necesitan compartir información acerca de la identidad y circunstancia del usuario, conocida como "información de contexto".

¿J2EE = SunONE?

Hay que aclarar que en este análisis se menciona la plataforma SunONE como referencia, sin embargo existen muchos paquetes de software de diversos proveedores que permiten implementar una arquitectura de servicios Web similar. SunONE es sólo una implementación de soluciones basadas en Java 2 Enterprise Edition (J2EE) y servirá como referencia y en representación de estas últimas. Dicho en otras palabras: SunONE es la implementación de J2EE que ofrece Sun Microsystems.

Construyendo Servicios Web con tecnologías de mayor aceptación

Los Servicios Web son en realidad simples interfaces basadas en XML de sistemas de servicios y aplicaciones de negocios, y son realmente viejas tecnologías con sombrero nuevo. Una forma para desarrollar Servicios Web es la siguiente:

- Un proveedor crea, ensambla y hace funcionar un Servicio Web empleando el lenguaje de programación, middleware y plataforma de su preferencia.
- El proveedor define el Web Service en un WSDL (Lenguaje Descriptor de Servicios Web). Un documento WSDL describe un Servicio Web a otros.
- El proveedor registra el servicio en registros UDDI (Descripción Universal, Descubrimiento e Integración). UDDI proporciona a los desarrolladores el repositorio dónde publicar sus Servicios Web, lo cual permite a su software buscar servicios ofrecidos por otros.
- Un usuario prospecto encuentra el servicio buscando en un registro UDDI.
- La aplicación del usuario se conecta al Servicio Web e invoca operaciones del servicio usando SOAP (Protocolo Simple de Acceso a Objetos). SOAP ofrece un formato XML para representar parámetros y regresar valores sobre HTTP. Es el protocolo de comunicación que todos los Servicios Web emplean.

Las tecnologías mencionadas anteriormente son sólo suficientes para Servicios Web simples. Para poder crear servicios más complejos que requieran transacciones, múltiples peticiones, esquemas, y flujo de documentos, se deberá usar una estructura más compleja, como por ejemplo ebXML. Sin embargo es importante el saber que SOAP, XML, UDDI y WSDL son tecnologías que han sido ampliamente aceptadas y las grandes compañías están trabajando con ellas para establecer y adoptar un verdadero estándar.

El acercamiento a Servicios Web de J2EE y MS .NET

Los Servicios Web deben ser confiables, de alta disposición, tolerantes a fallas, escalables y de gran desempeño, para lograrlo se debe hacer uso de un gran número de tecnologías y de esfuerzo.

La visión compartida entre J2EE y .NET es que existe un gran número de módulos o piezas de construcción que se enlazan para construir Web Services, tales como interoperabilidad XML, balanceo de carga, persistencia y manejo de transacciones. En lugar de escribir uno mismo todas esas piezas de construcción, uno puede escribir una aplicación que corre dentro de un contenedor que provee estos complicados servicios. Este paradigma le permite a uno el especializarse en lo que le importa, lo que le corresponde. Uno se preocupa por entender el problema del negocio y dejar al contenedor la carga de esas piezas de construcción. Veamos ahora con detalles cada visión.

3.2.1. La visión de J2EE

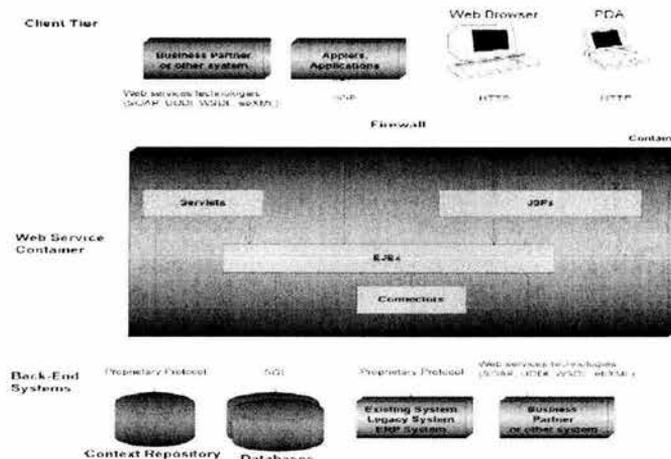
Java 2 Enterprise Edition fue diseñado para simplificar problemas complejos con el desarrollo, puesta en marcha y administración de sistemas empresariales multicapa. J2EE es un estándar en el mercado y es el resultado de la iniciativa de muchas industrias encabezadas por SUN Microsystems. Es importante señalar que J2EE es un estándar y no un producto. El objetivo de J2EE es dar a los clientes la oportunidad de elegir productos y herramientas de diversos vendedores. SUN colaboró con otros vendedores de plataformas eBusiness, tales como BEA, IBM y Oracle en la definición de J2EE. SUN entonces inició el Java Community Process (JCP) para solicitar nuevas ideas que mejoraran J2EE con el tiempo.

Java: El fundamento de J2EE

La arquitectura J2EE está basada en el lenguaje de programación Java. Lo excitante de éste lenguaje de programación, es que permite a una organización el escribir su código una vez y ejecutarlo en cualquier plataforma. Los desarrolladores escriben código fuente en Java, el código Java es compilado en *bytecode*, cuando el código está listo para correr, Java Runtime Environment (JRE) interpreta este *bytecode* y lo ejecuta. J2EE es una aplicación de Java, y como tal, los componentes de J2EE son transformados en *bytecode* y ejecutados por un JRE. Inclusive los contenedores de aplicaciones están típicamente escritos en Java.

J2EE y los Servicios Web

J2EE puede ser usado para construir sitios Web tradicionales, componentes de software o aplicaciones empaquetadas. Últimamente ha sido también extendido para soportar la construcción de Servicios Web basados en XML, estos servicios pueden interoperar con otros Servicios Web que pudieran o no haber sido escritos en el estándar J2EE. El modelo de desarrollo de Servicios Web J2EE se muestra en la siguiente figura:



La Visión de SunONE

Análisis

La aplicación J2EE se encuentra dentro de un contenedor, el cual provee servicios necesarios para una aplicación empresarial, tales como transacciones, seguridad y servicios de persistencia.

La capa de negocio desempeña procesamiento del negocio y lógica de información. En aplicaciones J2EE de gran escala, la lógica de negocio es construida empleando componentes Enterprise Java Beans (EJB). Esta capa desempeña procesos de negocio, se conecta a bases de datos usando Java Database Connectivity (JDBC) o SQLJ, conecta a sistemas existentes empleando Java Connector Architecture (JCA).

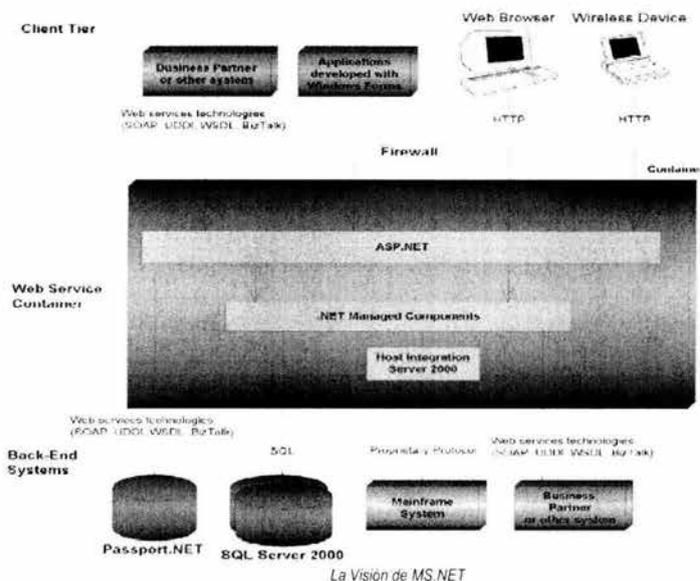
Los socios de negocios pueden conectar a otros socios de negocios usando tecnologías de Servicios Web (SOAP, UDDI, WSDL, ebXML) a través del API de Java para XML (JAX). Un servlet, el cual es un objeto Java, puede aceptar peticiones de Servicios Web de parte de socios de negocio. El servlet usa el API JAX para realizar operaciones Web Services.

Clientes delgados tradicionales tales como applets o aplicaciones se conectan directamente a la capa EJB a través de Internet Inter-ORB Protocolo (IIOP) en lugar de Servicios Web, esto debido a que generalmente los clientes delgados están escritos por la misma organización que desarrolló la aplicación J2EE.

Navegadores Web y dispositivos inalámbricos se conectan a Java Server Pages (JSPs) los cuales presentan interfaces de usuario en HTML, XHTML, o WML.

3.2.2. La visión de Microsoft .NET

Microsoft .NET es una colección de productos que permite a las organizaciones construir Servicios Web inteligentes y de clase empresarial. Notemos la diferencia: .NET es una estrategia de productos, mientras que J2EE es un estándar a los productos desarrollados. Microsoft .NET es en gran parte una reescritura de Windows DNA, el cual era la plataforma previa de MS para desarrollar aplicaciones empresariales y que incluía tecnologías como Microsoft Transaction Server (MTS) y COM+, Microsoft Message Queue (MSMQ), y la base de datos Microsoft SQL Server. El nuevo framework .NET reemplaza estas tecnologías e incluye una capa de Servicios Web así como un soporte a lenguaje mejorado. El modelo de desarrollo de Servicios Web con .NET se muestra en la figura de abajo.



La aplicación .NET se encuentra dentro de un contenedor, el cual provee los servicios necesarios para aplicaciones empresariales tales como transacciones, seguridad y servicio de mensajes.

La capa de negocios de la aplicación .NET es construida usando componentes administrados .NET. Esta capa desempeña procesos de negocios y lógica de información. Se conecta a bases de datos usando Active Data Objects (ADO.NET), COM Transaction Integrator (COM TI).

Socios de negocios pueden conectar con la aplicación .NET a través de tecnologías de Servicios Web (SOAP, UDDI, WSDL, BizTalk).

Cientes delgados tradicionales, navegadores Web y dispositivos inalámbricos se conectan a Active Server Pages (ASP.NET) el cual presenta al usuario interfaces en HTML, XHTML o WML.

3.2.3. J2EE y .NET Frente a Frente

Para ayudarnos a entender ambos modelos, veamos las características tecnológicas de J2EE y .NET. La siguiente tabla nos muestra tales características.

Característica	J2EE	.NET
Tipo de tecnología	Estándar	Producto
Vendedores de middleware	30+	Microsoft
Intérprete	JRE	CLR
Páginas Web dinámicas	JSP	ASP.NET
Componentes Middle-Tier	EJB	Componentes .NET
Acceso a Base Datos	JDBC, SQLJ	ADO.NET
SOAP, WSDL, UDDI	Si	Si
Características del middleware (seguridad, transacciones, etc.)	Si	Si

Análisis comparativo

Características de tiempo a mercado

Al desarrollar una solución comercial hoy día, hacerla disponible en unos cuantos meses es toda una eternidad. Perder una pequeña ventana de oportunidad es la diferencia entre una corporación que es la primera en el mercado y una corporación que debe recoger lo que sobra por años. Una manera de apresurar ese tiempo a mercado es eligiendo una plataforma de Servicios Web que permita un desarrollo rápido de aplicaciones, esto facultará a los desarrolladores para escribir y mantener el código de manera rápida, reduciendo el tiempo de desarrollo. Ambas plataformas, J2EE y .NET proveen mecanismos en tiempo de ejecución que aíslan a los desarrolladores de software de dependencias particulares. J2EE y .NET ofrecen intermediación a nivel de lenguaje por medio de JRE y CLR.

J2EE ofrece muchas características que aceleran el tiempo a mercado que no se encuentran en .NET. Por ejemplo, los servicios de manejo de estado permiten a los desarrolladores el escribir menos código y no preocuparse sobre como manejar el estado de una aplicación, resultando en un grado más alto de desarrollo de aplicaciones. Estos servicios permiten construir componentes que mantienen su propio estado. Los servicios de persistencia (entity beans) permiten a los desarrolladores crear aplicaciones sin tener que escribir código para controlar el acceso lógico a los datos, resultando en un código más limpio, independiente de la base de datos y mucho más fácil de construir y mantener.

Microsoft .NET ofrece una variedad de características tiempo a mercado que no se encuentran en J2EE. La más notable es ASP.NET que es independiente del dispositivo cliente y permite que las interfaces de usuario sean renderizadas en función del dispositivo del usuario sin tener que reescribir el código. Microsoft también tiene Componentes Queue que son superiores a los Message Driven Beans. Cabe señalar que Microsoft ha optado por remover características comunes en sistemas empresariales como servicios de estado y transacciones simples.

En conclusión, la habilidad para lograr el desarrollo rápido de aplicaciones ofrecido por J2EE y .NET es definitivamente diferente, sin embargo son comparables. Las diferencias son mínimas y resulta difícil el argumentar cual es mejor o cual ofrece mayores prestaciones.

Solución de un único vendedor

Al construir Servicios Web, en general se debiera preferir el tener una solución de un solo vendedor, ya que usualmente esto resulta más fácil, ínter operable y menos propenso a errores que una solución que dependa de integrar productos de dos o mas vendedores.

Una de los puntos fuertes de J2EE es que tiene una amplia variedad de herramientas, productos y aplicaciones en el mercado, los cuales proveen más funcionalidad en total que lo que cualquier vendedor pudiera ofrecer. Sin embargo esta fortaleza también es una debilidad ya que dichas herramientas J2EE son en algunas ocasiones no interoperables debido a imperfecciones en la portabilidad, lo que ocasiona que se deba realizar alguna especie de *hacking* a bajo nivel. Grandes vendedores como IBM, Oracle, BEA e iPlanet ofrecen cada uno soluciones para Servicios Web completas.

.NET ofrece una solución completa de parte de un único vendedor, Microsoft. Esta solución podría carecer de las características más sobresalientes que J2EE tiene, pero en general, la visión completa sobre Servicios Web de Microsoft es igual en alcance que cualquiera de los grandes vendedores de J2EE.

Soporte a Sistemas existentes

Las grandes compañías tienen código existente que ha sido escrito en una gran variedad de lenguajes sobre una gran variedad de sistemas como son COBOL, C++, SAP, BASIC, FORTRAN, SAP, etc. Es vital entonces para esas corporaciones el tener una ruta eficiente y rápida para preservar y volver a usar esas inversiones, después de todo, es ilógico pensar que las grandes compañías tendrían los fondos y el tiempo necesarios para reinventar todos sus sistemas existentes. Está integración es una de las tareas más demandantes cuando se trata de construir Servicios Web. Hay muchas formas de lograr esa integración usando J2EE:

- El Servicio para Mensajería de Java (JMS) para integrar con sistemas de mensajería existentes.
- Servicios Web para integrar con cualquier sistema.
- CORBA para crear interfaces a código escrito en otros lenguajes que pudieran existir en maquinas remotas.
- JNI para leer librerías nativas y llamarlas localmente.
- Java Connector Architecture (JCA), la característica más importante de J2EE y que permite conectar sistemas existentes tales como SAP, COBOL, etc.

.NET también ofrece integración a través del Host Integration Server 2000 y COM Tansaction Integrator (COM TI) para la colaboración de transacciones de sistemas mainframe.

Análisis

También tenemos Microsoft Message Queue (MSMQ), que puede integrar sistemas actuales usando IBM MQSeries, así como BizTalk Server 2000, que puede ser usado para integrar sistemas basados en protocolos Bussines to Bussines (B2B). En conclusión, las características para integración ofrecidas por J2EE son superiores a las ofrecidas por .NET.

Percepción del mercado

La realidad nos dice que las buenas tecnologías rara vez son exitosas en el mercado simplemente por que son buenas tecnologías, usualmente es la tecnología con el mejor mercadeo la que gana.

J2EE es una plataforma extremadamente bien presentada a mercado debido a que es vendida por una industria completa de más de 50 vendedores, lo que le da una gran percepción de mercado.

La fortaleza de .NET radica en el hecho de que Microsoft sabe muy bien como mercadear una plataforma. Tienen a su mejor equipo trabajando en la promoción de .NET y la ventaja de que ellos anunciaron su estrategia de Servicios Web antes que los desarrolladores de J2EE, atrayendo así a una gran parte del mercado. Parece ser que Microsoft .NET tiene una ventaja sobre J2EE en cuanto a la percepción del mercado.

Madurez de la plataforma

Las organizaciones que adopten una plataforma para Servicios Web deben considerar su madurez, ya que una plataforma no tan madura o de primera generación, es más propensa a errores y problemas. J2EE es una solución que tiene ya tiempo en el mercado. Actualmente existen en el mundo real muchas aplicaciones J2EE en línea, corriendo y resolviendo gran cantidad de problemas empresariales de misión crítica, sin embargo, si miramos más allá de la superficie hay algunas áreas identificables de riesgo donde J2EE carece de madurez:

- La persistencia automática que proporciona EJB es aún inmadura.
- La Arquitectura de Conectores de Java (JCA) es nueva.
- Todo el soporte a Servicios Web es nuevo.

Para Microsoft .NET la historia es un poco diferente. Partes de .NET están basadas en Windows DNA, el cual también corre en una gran variedad de sitios Web de misión crítica y disfruta de cierto éxito, sin embargo:

- Con el nuevo CLR, una considerable proporción de la base de la plataforma .NET ha sido sustancialmente rescrita, así que la plataforma en sí misma está en etapa Beta.
- C# es un lenguaje nuevo.
- Todo el soporte de Servicios Web es nuevo.

En conclusión, consideramos que J2EE es la plataforma más madura, a pesar de los componentes de reciente aparición.

Si bien es cierto que ciertas características de J2EE son nuevas y riesgosas, también es cierto que la plataforma .NET es demasiado nueva en sus bases y que el lenguaje C# es también muy nuevo, representando mayores riesgos que los implicados por J2EE.

Soporte de lenguaje

J2EE promueve un cómputo centrado en Java, de forma que todos los componentes que conforman una aplicación (tales como componentes EJB y servlets) deben estar escritos en el lenguaje Java. Para usar J2EE uno se debe comprometer a codificar por lo menos algunos de los sistemas usando el lenguaje de programación Java. Otros lenguajes pueden ser empleados y empataados en una solución J2EE a través de Servicios Web, sin embargo, estos lenguajes no pueden ser entre mezclados con el código Java.

En comparación, .NET soporta desarrollo en cualquier lenguaje que las herramientas de Microsoft soporten, esto debido al nuevo CLR (Common Language Runtime). Con la excepción de Java, todos los grandes lenguajes estarán soportados, así como el nuevo lenguaje orientado a objetos, C#, que es equivalente a Java. Todos los lenguajes soportados por el CLR son interoperables de modo que una vez traducidos a MSIL (Microsoft Intermediate Language), son efectivamente un lenguaje común. Un único componente .NET puede entonces ser escrito en diferentes lenguajes.

El soporte a diversos lenguajes que Microsoft ha introducido con el CLR es una innovación para los negocios. Es claramente una característica que aventaja a .NET sobre J2EE. Sin embargo, ¿es CLR una ventaja para los negocios?: CLR podría representar una elección de diseño pobre si más de un lenguaje es empleado, estas son las razones:

Riesgo. Muchos sistemas existentes están fuertemente acoplados y guardan relaciones muy complicadas. Romper con tal integración en los sistemas existentes es una propuesta arriesgada. El viejo adagio que dice "Si no está roto, no lo arregles" se aplica aquí.

Mantenimiento. Especulamos que una combinación de lenguajes corriendo en el CLR podría llevar a una sopa de letras de código que sería de un mantenimiento muy difícil. Se requerirían expertos en diversos lenguajes, llevando a un incremento en el costo para capacitar a los desarrolladores, lo que a su vez implica un costo total del sistema más elevado.

Construcción del conocimiento. Con una combinación de código de lenguajes los desarrolladores no podrían compartir sus mejores habilidades y prácticas. Se crearían grupos separados de desarrollo con lo que se rompería la comunicación y reduciría la productividad del equipo.

Transferencia de habilidades. Mientras que desarrolladores empleando diferentes lenguajes podrían codificar rápidamente un sistema .NET usando VB.NET y C#, ¿que pasaría si de pronto los nuevos desarrolladores C# deciden abandonar la organización? Hay dos opciones: entrenar a los desarrolladores de VB.NET para entender y escribir código con C#, o contratar a desarrolladores C# que no conocen absolutamente nada acerca del código base. El resultado es un costo más elevado en el desarrollo y propiedad del sistema.

Análisis

En la mayoría de los casos, creemos que es un mucho mejor diseño el estandarizar a un único lenguaje y tratar a los sistemas como uno solo y completo, e integrarlos a través de los API's correspondientes, lo que se puede lograr tanto con J2EE como en .NET. En este caso, la ventaja de .NET sobre J2EE es una característica que idealmente no se debería utilizar.

Migración de una plataforma previa

Para organizaciones que tienen sistemas existentes usando tecnologías basadas en J2EE o Windows DNA, un interesante punto de discusión es la facilidad para migrar de una plataforma previa a una nueva.

J2EE no representa muchos problemas de migración. Como se mencionó antes: JCA y Servicios Web son tecnologías nuevas en J2EE y requerirán nuevo código, pero esas son dificultades menores.

Aunque Microsoft.NET está basado en MTS y COM+, hay preocupación en que la migración a .NET será más costosa comparado a J2EE. Primero que nada .NET está basado en el framework de "código administrado" que toma muchas ideas de COM+ y MTS, pero es enteramente una nueva infraestructura basada en un código base completamente nuevo: CLR. Esto impone ciertas restricciones importantes al momento de migrar.

Por ejemplo, para acomodar un Sistema de Tipos Común (CTS) el cual estandariza los tipos de datos usados entre lenguajes, Microsoft ha desechado por completo los tipos de datos básicos de Visual Basic. En consecuencia, el código que dependa en esos tipos básicos de Visual Basic romperán con la migración y no existe actualmente herramienta de migración para ello. Como se puede ver, la migración no es libre ni gratuita para la plataforma .NET. Pero debemos entender que con la innovación que representa CLR, es un paso necesario que deben dar sus clientes si desean involucrarse con la nueva plataforma. Así que concluimos que la migración en J2EE es mucho más fácil comparada con .NET.

Portabilidad

Una diferencia clave entre J2EE y .NET es que J2EE no está ligado a plataforma alguna y corre en una gran variedad de hardware y sistemas operativos, tales como Windows, Solaris, Linux, BSD y sistemas mainframe. Tal portabilidad es una absoluta realidad hoy día debido a que el Java Runtime Environment (JRE), en el cual J2EE está basado, se encuentra disponible en cualquier plataforma.

Hay también, un segundo y más debatible aspecto de la portabilidad. J2EE es estándar de forma que soporta una variedad de implementaciones tales como BEA, IBM y Sun. El peligro radica en que los vendedores pueden no apegarse al estándar propuesto por Sun, provocando un sacrificio en la portabilidad. Para evitar tal situación, Sun ha construido un conjunto de pruebas de compatibilidad J2EE que aseguran que las plataformas J2EE cumplen con el estándar. Este conjunto de pruebas son críticas por que aseguran la portabilidad de las aplicaciones. Puede ser muy ingenuo pensar que la portabilidad de un sistema complejo se lleva a cabo sin complicaciones y de manera gratuita, pero hay que apuntar que la portabilidad es exponencialmente más barata y fácil con la plataforma J2EE y su conjunto de pruebas de compatibilidad.

En comparación, .NET sólo corre en sistemas Operativos Windows, el hardware que éste soporta y su ambiente .NET. No se puede hablar de portabilidad entre plataformas.

En conclusión, si la portabilidad es una consideración importante del sistema, J2EE es entonces la mejor opción y esto es la diferencia más crítica entre J2EE y .NET.

Soporte para Servicios Web

El futuro de la colaboración eBusiness son sin duda los Servicios Web. Para las organizaciones que están en persecución de una estrategia de Servicios Web o están preparándose para el futuro de los Servicios Web, su arquitectura base de eBusiness debe tener un fuerte soporte de Servicios Web.

Hoy día, J2EE soporta Servicios Web a través del API de Java para XML (JAXP). Esta API permite a los desarrolladores ejecutar cualquier operación de Servicios Web con análisis sintáctico manual de documentos XML. Hay disponibles una gran variedad de herramientas de terceros compatibles con J2EE que permiten el rápido desarrollo de Servicios Web y soportan la generación automática de interfaces y descriptores. Hay varias implementaciones SOAP que soportan Java, y casi todas están construidas sobre J2EE (servlets o JSP's). Hay también varias implementaciones del API de UDDI y la mayoría de ellas soportan Java.

La versión beta de Microsoft.NET también permite a las organizaciones el construir Servicios Web. Las herramientas que se incluyen ofrecen desarrollo rápido de Servicios Web, también con generación automática de descriptores.

En conclusión, se pueden desarrollar Servicios Web para la plataforma J2EE usando JAXP, sin embargo esta no es la forma ideal de hacerlo ya que requiere mucha intervención manual. Una alternativa es emplear soluciones de terceros para acelerar el desarrollo, lo que representa tener que adquirir librerías para nuestra aplicación. Con .NET se pueden desarrollar Servicios Web empleando la versión parcial de .NET. Sin embargo, debido a que se trata de una versión Beta, no representa una plataforma completamente seria. Otro punto en contra de .NET, es que no soporta Servicios Web verdaderos, pues aún carece de soporte para ebXML, un importante estándar para eBusiness que tiene amplia aceptación en el mundo.

Herramientas

El portafolio de productos de J2EE incluye Forte, un IDE de desarrollo modular basado en Java que data de tiempos anteriores a J2EE y .NET. Si los desarrolladores lo prefieren pueden emplear otros IDE's de desarrollo Java que son gratuitos como Visual Café de WebGain, VisualAge de IBM, JDeveloper de Oracle y algunos más. Numerosas herramientas y código open-source también están disponibles.

Microsoft siempre ha sido un fuerte vendedor de herramientas y eso no ha cambiado. Como parte del lanzamiento de .NET, Microsoft ofreció una versión Beta del conjunto de desarrollo Visual Studio.NET que soporta todos los lenguajes ya previamente soportados por VisualStudio exceptuando a Java. En su lugar es soportado C# (el nuevo lenguaje orientado a objetos de Microsoft) y herramientas nuevas como Web Forms.

En conclusión, Microsoft es el claro ganador en lo que se refiere a herramientas de desarrollo. Aunque la funcionalidad que ofrece J2EE como conjunto de aplicaciones es superior, es cierto también que estas no son 100% interoperables, debido a que no se originan de un único vendedor y se vuelve necesario hacer ciertos *trucos* (o *hacking*) de bajo nivel.

Contexto compartido

Un elemento clave de Servicios Web inteligentes es el *contexto compartido*. La visión del contexto compartido es que uno escriba información confidencial una sola vez y que esa información esté automáticamente disponible a todos los Servicios Web que tengan permiso legítimo de accederla. La información está bajo nuestro control, y no bajo el control de los Servicios Web y es protegida empleando reglas que uno puede definir.

La visión de J2EE de Sun está en un conjunto descentralizado y distribuido de servicios de contexto compartido que viven en Internet. Uno puede consultar el Servicio Web de este servicio de contexto compartido al conectarnos para determinar los permisos de acceso a la información.

Hoy día, los desarrolladores están ocupados en crear servicios de contexto compartido al escribir un servlet que se exponga a sí mismo como un Servicio Web.

En comparación, Microsoft.NET logra el contexto compartido a través del servicio de Passport.NET que es un repositorio centralizado mantenido por Microsoft que contiene información de la identidad del usuario.

Es la punta de lanza de los servicios Hailstorm de Microsoft e implica una visión de Servicios Web centralizados.

En conclusión, ambas plataformas soportan contexto compartido y cada una tiene sus pros y sus contras.

La diferencia mas notable es que el acercamiento de J2EE para un contexto compartido estándar, resucitará un mercado de repositorios distribuidos en Internet, mientras que la solución de Microsoft.NET es un acercamiento a un único repositorio central.

Las ventajas del acercamiento de J2EE son:

- Cada repositorio de contexto compartido puede ser especializado para diferentes necesidades. Por ejemplo, puede haber un repositorio para información médica, otro para información financiera, etc. Es muy poco probable que un acercamiento de un único repositorio como lo es Passport.NET fuera lo suficientemente especializado para cubrir todas las necesidades.
- No hay un efecto de "big brother". Negocios e individuos no necesitan confiar su información a cualquier firma individual. Se pueden emplear repositorios de clientes o socios confiables.
- No hay un único punto o falla.

- La información no es controlada por una única organización. Es difícil el imaginar que sólo un vendedor controle el servicio de identificación de todas las organizaciones.

Las ventajas del acercamiento de .NET son:

- No hay duda sobre cual es el repositorio oficial de contexto compartido. Hay sólo un único lugar donde encontrar información confiable sobre la identidad.
- Passport es un sistema activo ya establecido.
- Hasta que J2EE no estandarice un esquema o un API para acceder a servicios de contexto compartido, éstos no tienen una aplicación real.

Costo del sistema

Una amplia variedad de implementaciones basadas en la arquitectura J2EE están ya disponibles, con precios que varían dramáticamente, permitiendo a las corporaciones el elegir la plataforma que más cubra su presupuesto y nivel de servicio.

Los costos están por lo regular en las cifras de los miles de dólares por procesador; también existen las soluciones gratuitas de código abierto, aunque muchas empresas las consideran menos confiables. Al momento de escribir esta información, Microsoft no tenía aún información sobre los costos de la plataforma .NET.

En lo que se refiere al hardware, J2EE soporta UNIX, Linux, sistemas mainframe y virtualmente cualquier plataforma compatible con Java; ambos (.NET y J2EE) soportan la plataforma Win32, la cual es por lo general la alternativa menos costosa.

La ventaja la tiene J2EE, debido a que se pueden construir soluciones tanto de bajo costo como sistemas de alto costo. Se tiene la opción de crear un sistema empleando iPlanet corriendo en un servidor SunFire, o uno basado en JBoss corriendo en Linux sobre un Servidor con procesadores Intel. También hay que considerar que existe un buen conjunto de herramientas y servicios OpenSource que soportan Java. Adicionalmente hay que mencionar que el costo de la plataforma es siempre una pequeña parte del costo total del proyecto: no se puede considerar únicamente el precio de la plataforma y omitir otros costos como son el entrenamiento de los desarrolladores, y el costo del desarrollo e implementación.

Desempeño

Para que el sistema alcance un desempeño aceptable, es importante que la infraestructura sobre la que se construyen los Servicios Web nos permita construir componentes de alto rendimiento. El cuello de botella principal al construir cualquier sistema, es usualmente la integración con sistemas de bases de datos.

La razón es que la mayoría de las aplicaciones empresariales son sistemas que implementan más lógica de información que lógica de negocio; cualquier táctica posible para reducir la carga a la base de datos resultará en una ganancia significativa en el desempeño de nuestra aplicación.

Análisis

J2EE reduce el tráfico a la base de datos a través de dos técnicas:

Stateful business processing permite mantener el estado de un proceso de negocio en memoria, en lugar de escribir el estado a la base de datos en cada petición.

Long-term caching (proporcionado por algunas implementaciones) permite que la información de una base de datos se coloque en memoria por largos periodos de tiempo, en lugar de volver a leer la información de la base de datos en cada petición.

Estas dos características deben ser usadas con mucha precaución, pues podrían resultar en problemas si los desarrolladores no están correctamente entrenados para usarlas.

Esta es una diferencia fundamental entre las 2 arquitecturas: la ventaja de J2EE es que da a los programadores más control sobre servicios de más bajo nivel como lo es el manejo de estado. Desarrolladores bien educados pueden apoyarse en estas características para mejorar la calidad de su sistema.

En comparación, Microsoft.NET no ofrece estas características tácticas para mejorar el desempeño. No existe oportunidad para ganar rendimiento, pero al mismo tiempo se protege al sistema de la introducción de errores por manejo explícito de aspectos de bajo nivel, como lo es el estado de los componentes.

Escalabilidad

La escalabilidad es esencial cuando con el tiempo se requiere hacer crecer un sistema basado en Servicios Web. Una plataforma es escalable si un incremento en los recursos de hardware resulta en un correspondiente incremento (idealmente lineal) en la cantidad de usuarios soportados, manteniendo el mismo tiempo de respuesta. Tanto J2EE como .NET ofrecen incrementar el número de máquinas para incrementar los usuarios soportados; ambos proveen balanceo de carga que permite a un cluster de máquinas colaborar y soportar una mayor carga de usuarios con el tiempo. Una diferencia importante estaría en que .NET soporta únicamente sistemas basados en plataforma Windows, lo que usualmente significaría hardware a menor precio, pero en mayor cantidad que en una plataforma Unix. Esta multitud de máquinas podría resultar muy difícil de organizar, administrar y mantener.

Conclusiones

Sin duda alguna la batalla J2EE contra .NET será el gran espectáculo de la década en el mundo de la informática y el Internet. J2EE es un movimiento brillante de parte de los vendedores participantes, pero no debe ser visto como una iniciativa altruista ya que todos esos participantes están en busca de ganancias económicas y de establecer un arma efectiva contra Microsoft.

Por su parte, Microsoft.NET (que también está lejos de ser una iniciativa filantrópica) resulta ser una iniciativa monopolista disfrazada de altruista. Microsoft clama que .NET son Servicios Web abiertos e interoperables, cuando en realidad la propia Microsoft está ya desarrollando sus Servicios Web cerrados y propietarios. Microsoft muy seguramente incrementará los costos de su solución y alentará drásticamente la innovación en caso de lograr un monopolio de este mercado.

Ambas plataformas son útiles y ambas pueden ayudarnos a llegar al mismo destino. Para tomar una decisión sobre cual plataforma usar, se deben considerar muchos aspectos, como las habilidades actuales de los desarrolladores, los sistemas existentes, los sistemas actuales y los clientes potenciales.

Los argumentos a favor de ambas plataformas son:

- No importando la plataforma que se elija, los nuevos desarrolladores deberán ser capacitados (Java para J2EE, C# y CLR para .NET).
- Se pueden emplear ambas plataformas para desarrollar Servicios Web.
- Ambas plataformas ofrecen soluciones de bajo costo (Software Libre/Linux/PC, o .NET/Windows/PC).
- Ambas plataformas ofrecen soluciones de un único vendedor.
- La escalabilidad de ambas plataformas es teóricamente ilimitada.

Los argumentos a favor de .NET y en contra de J2EE son:

- .NET está siendo promovido por el mejor equipo de mercadotecnia de Microsoft.
- .NET liberó su soporte de Servicios Web antes que J2EE.
- .NET tiene un mejor historial en lo que respecta a ambientes de contexto compartido.
- .NET tiene un excelente historial de herramientas de desarrollo con Visual Studio.NET.
- .NET tiene un modelo de programación más simple y apropiado para desarrolladores de nivel intermedio.
- .NET ofrece neutralidad de lenguaje cuando se desarrollan nuevas aplicaciones eBusiness a través de su ambiente CLR, mientras que J2EE obliga a tratar otros lenguajes como aplicaciones separadas.
- .NET se beneficia de estar altamente entrelazado con el sistema operativo sobre el que corre.

Los argumentos a favor de J2EE y en contra de .NET son:

- J2EE está siendo promovido por una industria entera.
- J2EE es una plataforma ya probada con unos cuantos API's nuevos de soporte para Servicios Web, .NET por otro lado, es nuevo y con ello introduce riesgos como cualquier tecnología de primera generación.
- Sólo J2EE permite poner en funcionamiento Servicios Web reales hoy día.
- Código J2EE existente se traducirá a Servicios Web J2EE sin mayores modificaciones, lo cual no aplica para el código Windows DNA si quiere ser portado a .NET.
- Los Servicios Web .NET no son interoperables con estándares actuales de la industria. Su framework *BizTalk* tiene extensiones propietarias de SOAP y no soporta ebXML.
- J2EE es un modelo de programación más avanzado, apropiado para desarrolladores bien entrenados que desean construir modelos de objetos avanzados y tomar ventaja de sus características de desempeño.
- J2EE permite tomar ventaja del hardware con que se cuente.
- J2EE brinda neutralidad de plataforma, incluyendo Windows. También se obtiene una portabilidad bastante aceptable de forma relativamente sencilla.
- J2EE permite emplear cualquier sistema operativo, incluyendo Windows, UNIX, Linux, mainframe, etc.
- J2EE permite usar Java, el cual tiene una mayor participación, madurez y aceptación en el mercado que C#.
- No es recomendable usar otro lenguaje diferente a Java o C# para desarrollo de nuevas soluciones de misión crítica; los demás son versiones de lenguajes con parches para soportar objetos (como C, VB, COBOL, etc.).
- Más y más compañías están decidiéndose por J2EE debido a que no pueden controlar las plataformas que sus clientes prefieren. Quizá esto sea lo que haga que J2EE empiece a dominar en el mercado.

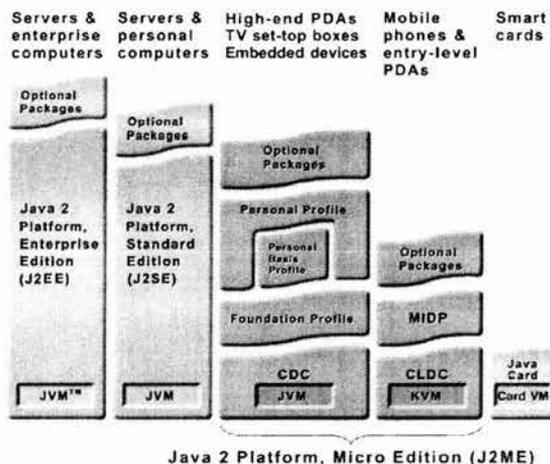
En conclusión, aunque ambas plataformas tendrán su propia participación en el mercado, creemos que la mejor ruta a seguir es la que nos ofrece J2EE. Preferimos emplear tecnologías basadas en J2EE para el desarrollo de nuestros Servicios Web por que consideramos que las ventajas que ofrece son mayores a aquellas que ofrece .NET.

3.3. La Plataforma J2ME para Dispositivos Portátiles

La plataforma Java 2 Micro Edition (J2ME) está basada en el lenguaje de programación Java y orientada a dispositivos inmersos y de consumo, como pueden ser: teléfonos celulares, PDA's, sistemas de información a bordo de vehículos y una amplia gama de dispositivos. Al igual que sus contrapartes empresarial (J2EE), de escritorio (J2SE) y de tarjetas inteligentes (Java Card), la plataforma J2ME es un conjunto de API's estándar de Java definidos por el programa "Java Community Process" formado por grupos de expertos que incluyen líderes productores de dispositivos, software y servicios. La plataforma J2ME brinda el poder y los beneficios de la plataforma Java en dispositivos portátiles, incluyendo una interfaz de usuario flexible, un modelo de seguridad robusto, amplio soporte para protocolos en red así como las provisiones necesarias para aplicaciones conectadas en red (on-line) y desconectadas de ella (off-line). Con la plataforma J2ME, las aplicaciones se escriben una vez y se pueden ejecutar en muchos dispositivos, son descargadas de la red dinámicamente y se explotan las capacidades nativas de cada dispositivo específico. La plataforma J2ME es incluida en millones de dispositivos, desde teléfonos móviles y PDA's hasta dispositivos en automóviles, además es soportada por proveedores líderes en proporcionar herramientas basadas en tecnología Java, y es usada por compañías en todo el mundo. En pocas palabras, ésta es la plataforma de elección para los dispositivos inmersos y de consumo en la actualidad.

La Plataforma J2ME

La plataforma J2ME define configuraciones, perfiles y paquetes opcionales como los elementos básicos para ensamblar un entorno de ejecución que satisfaga los diferentes requerimientos de la amplia variedad de dispositivos y mercados. Cada combinación, está optimizada para la memoria, el poder de procesamiento y las capacidades de entrada/salida de la correspondiente categoría de dispositivos. El resultado es una plataforma Java común que explota al máximo las capacidades de cada tipo de dispositivo para proporcionar una experiencia más completa al usuario.



Plataforma J2ME

Configuraciones

A la vez que J2ME supone un novedoso campo por desarrollar, de la misma manera introduce términos que debemos entender para asumir la arquitectura del sistema. El primer término que debemos asumir es el de *Configuración*. Una configuración es un conjunto mínimo de API's que son útiles para desarrollar aplicaciones para un conjunto definido de dispositivos. Son muy importantes, porque describen las funcionalidades mínimas requeridas para unos dispositivos determinados. Las configuraciones están formadas por una máquina virtual de Java y un conjunto mínimo de librerías de clases. Estas últimas proporcionan la funcionalidad básica para un conjunto de dispositivos en particular que comparten características similares de, por ejemplo, conectividad en red y perfil de memoria. Actualmente existen 2 configuraciones J2ME: la "Connected Limited Device Configuration" (CLDC) y la "Connected Device Configuration" (CDC).

CLDC

CLDC es un estándar que SUN ha especificado. Según se aplique J2ME a nuevas familias de dispositivos, SUN especificará nuevos estándares adecuados a cada familia. CLDC es la más pequeña y básica las configuraciones, y está diseñada para dispositivos con conexión intermitente a la red, procesadores lentos y memoria limitada (como teléfonos móviles, radios de 2 vías y PDA's de bajo poder). Estos dispositivos típicamente tienen procesadores de 16 o 32 bits y entre 128kB y 512kB de memoria disponible para la implementación de la plataforma Java y sus aplicaciones asociadas.

Extendiendo un poco el concepto de CLDC, este estándar especifica los siguientes aspectos de la programación wireless: El subconjunto del lenguaje java que puede ser usado, el subconjunto de funciones de la Máquina Virtual Java, las API's fundamentales para este tipo de desarrollo y los requerimientos de hardware de los dispositivos móviles enfocados a CLDC.

Algunas características de Java han sido deshabilitadas y la razón es sencilla: los dispositivos móviles tienen capacidad limitada. Esto a su vez, limita algunos aspectos de Java y de la Máquina Virtual por igual. La función más importante que cumple CLDC es indicar un conjunto de API's que debe incorporar cualquier dispositivo. Finalmente, CLDC establece los mínimos de hardware requeridos para J2ME. Estos son los siguientes:

- 160kB de memoria disponible para Java
- Procesador de 16-bit
- Bajo consumo energético
- conexión a una red (a una velocidad típica de 9600bps pero puede ser menor).

Examinando estos requerimientos más a fondo, es conveniente saber qué significa el requerimiento de 160kB de memoria disponible para java. CLDC especifica que esta cantidad está compuesta de 128kB de memoria no volátil para la Máquina Virtual Java y las API's de CLDC; otros 32kB de memoria volátil están destinados a el entorno de ejecución de Java.

CLDC y Java

Veamos que restricciones le impone el CLDC al lenguaje de programación Java. La primera limitación es la falta de soporte para los números de punto flotante, de manera que no ofrece mecanismos para realizar operaciones matemáticas muy complejas. La limitación se impone porque los dispositivos carecen de hardware para estas operaciones y hacerlo vía software sería cargarlos por encima de sus posibilidades. La siguiente restricción es la eliminación del método `Object.finalize()`. Este método es llamado para que un objeto sea borrado de memoria y CLDC no lo soporta ni lo requiere. Una restricción a tener en cuenta es la limitada capacidad de CLDC para el manejo de excepciones. CLDC maneja un número limitado de excepciones y delega el resto a las API's específicas de cada familia de dispositivos.

CLDC y Seguridad

Al igual que la versión estándar de Java, J2ME tiene su propio modelo de seguridad. Si se está habituado a desarrollar applets se estará familiarizado con el modelo "sandbox" de seguridad en Java (del que se ha hablado anteriormente).

La filosofía de este modelo de seguridad es dotar a los desarrolladores de grandes posibilidades para hacer aplicaciones potentes, pero por otra parte minimizar los riesgos que supone un dispositivo que ejecuta aplicaciones descargadas de la red. Esto define las líneas maestras del modelo sandbox de seguridad:

- Las aplicaciones Java deben ser verificadas como válidas.
- Sólo se permite el uso de API's autorizadas por CLDC
- No está permitido cargar clases definidas por el usuario.
- Sólo las características nativas que entren dentro del CLDC pueden ser accedidas.

Se supone que una capa adicional, que está definida en una configuración, impondrá nuevas restricciones.

CDC

CDC está diseñado para dispositivos que tienen más memoria, procesadores más rápidos y una conexión a red de banda ancha, como pueden ser gateways residenciales, sistemas de navegación a bordo de vehículos y PDA's más potentes.

CDC incluye una máquina virtual de Java completa y un subconjunto mucho mayor de clases que la configuración CLDC. Esto implica que los dispositivos que soportan esta configuración, tienen procesadores de 32 bits y un mínimo de 2MB de memoria disponible para la implementación de la plataforma Java y sus aplicaciones asociadas.

Perfiles

Con el objetivo de proporcionar un entorno de ejecución completo para alguna clase de dispositivos específica, las configuraciones deben estar combinadas con un conjunto de API's de un nivel superior (perfiles), que definen más concretamente el modelo del ciclo de vida de una aplicación, la interfaz de usuario y el acceso a las propiedades particulares del dispositivo.

MIDP: Perfil de Dispositivo Móvil de Información (Mobile Information Device Profile)

El MIDP, está diseñado para teléfonos móviles y PDA's de bajo poder. Ofrece la funcionalidad básica para aplicaciones móviles, incluyendo la interfaz del usuario, la conectividad a la red, almacenamiento local de datos y administración de las aplicaciones. Combinado con CLDC, MIDP proporciona un entorno de ejecución Java completo que explota las capacidades de dispositivos de bolsillo, y minimiza tanto el uso de memoria como de energía.

CLDC es la configuración básica de J2ME. MIDP lleva CLDC más allá y añade nuevos requerimientos y API's obligatorios para dispositivos MIDP. Los requerimientos de memoria de MIDP son:

- 128kB de memoria no volátil para las librerías MIDP API.
- 32kB de memoria volátil para el sistema Java runtime.
- 8kB de memoria no volátil para datos de aplicación persistente.

Los requerimientos de entrada para MIDP exigen la existencia de un teclado o una pantalla táctil o ambos. No se exige ratón (ya que son raros en un teléfono móvil). Los requerimientos de salida para MIDP son algo más importantes, porque la pantalla es una de las restricciones mayores de los dispositivos móviles. MIDP exige al menos una pantalla de 97 x 54 píxeles (ancho x alto) con 1-bit de profundidad de color (blanco y negro). El ratio de salida debe ser de preferencia de 1:1. MIDP tiene también requerimientos de red. El mínimo soporte de red exigido es disponer de una conexión inalámbrica de 2 sentidos. MIDP no establece ninguna obligación respecto a qué sistema operativo debe tener el dispositivo (esto es posible gracias a que Java es multiplataforma). Sin embargo, MIDP sí espera ciertos requerimientos mínimos: un kernel capaz de manejar el hardware a bajo nivel, un mecanismo que lea y escriba en memoria persistente, un mecanismo de temporización para establecer mediciones de tiempo y dotar de información de hora y fecha a los datos persistentes, acceso de lectura y escritura hacia la conexión inalámbrica, acceso a la entrada por teclado o pantalla, soporte mínimo para mapas de bits y un mecanismo que controle el ciclo de vida de una aplicación

FP: Perfil Básico (Foundation Profile)

Los perfiles CDC están organizados en capas, de forma que puedan ser extendidos de acuerdo a las necesidades de funcionalidad de cada tipo de dispositivo. El perfil básico (foundation profile, FP) es la implementación más básica de CDC con soporte para red que puede ser usado con aplicaciones muy inmersas que no usen una interfaz de usuario.

PP: Perfil Persona (Personal Profile)

El perfil personal (personal profile, PP) es un perfil de CDC orientado a dispositivos que requieren soporte completo para applets o para una interfaz gráfica de usuario, como puede ser un PDA muy potente, algún dispositivo de tipo "communicator", o consolas de juegos. Este perfil incluye las librerías completas de AWT de Java y permite correr fácil y correctamente applets a través de Web diseñados para sistemas de escritorio.

PBP: Perfil Personal Base (Personal Basis Profile).

El perfil de base personal (PBP) es un subconjunto de PP que proporciona un entorno de ejecución de aplicaciones para dispositivos conectados a la red que soportan un nivel muy básico de presentación gráfica o que requieren el uso de herramientas y API's específicas. Estos dispositivos incluyen sistemas de navegación a bordo de vehículos y módulos de información. Tanto PP y PBP constituyen una capa construida por encima de CDC y FP.

Paquetes Opcionales

La plataforma J2ME puede ser extendida más allá al agregarle varios paquetes opcionales a CLDC, CDC y sus perfiles correspondientes. Creado para satisfacer necesidades muy específicas de los mercados, los paquetes opcionales ofrecen API's estándares que permitan usar tanto tecnologías existentes como las que apenas están surgiendo, como Bluetooth, Servicios Web, servicios de mensajería inalámbrica, multimedia y conectividad a bases de datos. Dado que los paquetes opcionales son modulares, los fabricantes de los dispositivos pueden incluirlos como sea necesario para explotar al máximo las prestaciones de sus productos.

J2ME es una tecnología de muy reciente aparición. Se puede prever el enorme mercado potencial de las comunicaciones a través de dispositivos portátiles inalámbricos. Indudablemente, las características naturales de Java (su modularidad, seguridad, flexibilidad, independencia de plataforma, etc.) hacen creer que será un lenguaje importante dentro de este sector.

3.4. Wireless: Un Mundo sin Cables

3.4.1. WLAN's y Tecnologías de Comunicación Inalámbricas

A medida que las comunicaciones han logrado prescindir de los cables y convertirse en gran parte en inalámbricas, las empresas y prestadores de servicios han visto surgir una amplia gama de posibilidades anteriormente inexistentes. Por ejemplo, la posibilidad de desplegar sus redes más rápidamente y con menor costo, y la presencia de aplicaciones móviles que integran una amplia gama de servicios. En México, al igual que en casi todos los países, los nuevos operadores deben construir en el corto plazo redes que les permitan competir con un ex monopolio que construyó su infraestructura en varias décadas. En este sentido, la comunicación inalámbrica se ha vuelto pieza clave para poder comercializar cuanto antes sus servicios de comunicación, sin depender por completo de la fibra óptica o el cobre.

El potencial de las comunicaciones inalámbricas ya se vislumbraba desde hace algunos años, y es por eso que desde su creación, en 1996, la Comisión Federal de Telecomunicaciones (COFETEL) ha dado a la licitación del espectro radioeléctrico el estatus de actividad prioritaria. En los últimos años han sido subastadas bandas de frecuencias para diversos servicios: radiolocalización de una y dos vías, enlaces de microondas (punto-punto y punto-multipunto en distintas frecuencias), radiocomunicación móvil terrestre, acceso fijo inalámbrico y telefonía celular PCS. Igual que en todo el mundo, la liberación de estas frecuencias se presenta como factor clave para el surgimiento de numerosos servicios inalámbricos.

Por comunicación inalámbrica debe entenderse aquella que no requiere de cables. La variedad más importante es la que se realiza haciendo uso de la radiocomunicación (aunque otra opción es la luminosa), y de ahí la gran importancia que tiene dividir y asignar en forma óptima el espectro radioeléctrico. Las comunicaciones inalámbricas ofrecen más seguridad al tener intrínsecamente una etapa de digitalización, menor costo por suscrito o por aplicación, se instalan más rápidamente y el costo de mantenimiento es más bajo.

Las comunicaciones inalámbricas son normalmente más fáciles y económicas de implementar, ya que se elimina mucho del trabajo de instalación y mantenimiento del cableado. Por otro lado, las redes inalámbricas tienen una desventaja frente a las alámbricas, y es que son más susceptibles a interferencias del medio ambiente. Aunque la opción inalámbrica es más rápida y económica en su construcción, las redes cableadas aún ofrecen velocidades de transmisión mucho más altas. Sin embargo, las redes inalámbricas tienen una ventaja adicional, la movilidad, y es eso lo que explica el auge que están mostrando en la actualidad.

WLAN: Redes Inalámbricas

WLAN son las siglas en inglés de Wireless Local Area Network. Redes de área local inalámbricas. Es un sistema de comunicación de datos flexible muy utilizado como alternativa a la LAN cableada o como una extensión de ésta. Utiliza tecnología de radio frecuencia que permite mayor movilidad a los usuarios al minimizarse las conexiones cableadas. Las WLAN han adquirido importancia en muchos campos incluido el de la medicina.

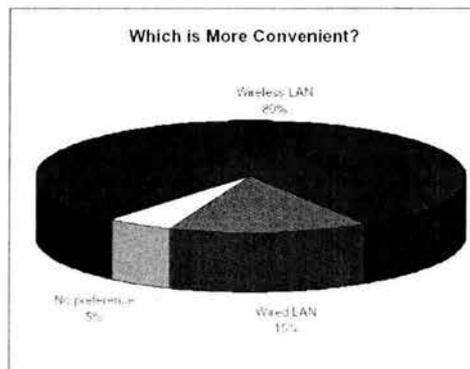
Como ejemplos de uso tenemos: ventas al por menor, almacenes, manufactura, etc. En estas aplicaciones, se transmite la información en tiempo real a un procesador central. Cada día se utilizan más este tipo de redes en un amplio número de negocios y se augura una gran extensión de éstas, permitiendo a las empresas tener un corto tiempo de *ROI* (Retorno de Inversión) y altas ganancias.

¿Por qué utilizar WLAN's?

Es clara la alta dependencia en los negocios de las redes de comunicación. Por ello la posibilidad de compartir información sin que sea necesario buscar una conexión física permite mayor movilidad y comodidad. Así mismo la red puede ser más extensa sin tener que mover o instalar cables. Respecto a la red tradicional, una red sin cables ofrece las siguientes ventajas:

- *Movilidad*: Información en tiempo real en cualquier lugar de la organización o empresa para todo usuario de la red. El que se obtenga en tiempo real supone mayor productividad y posibilidades de servicio.
- *Facilidad de instalación*: Evita obras para instalar cable por muros y techos.
- *Flexibilidad*: Permite llegar donde el cable no puede.
- *Reducción de costos*: Cuando se dan cambios frecuentes o el entorno es muy dinámico, el costo inicial (generalmente el más alto) de la red sin cable es significativamente más bajo que el de una red alamburada.
- La infraestructura de una red inalámbrica tiene mayor tiempo de vida y menor gasto de instalación y mantenimiento.
- *Escalabilidad*: El cambio de topología de red es sencillo y trata por igual a redes grandes o pequeñas.

La siguiente gráfica muestra los resultados de una encuesta realizada a varias empresas donde se les preguntó qué tipo de red era más conveniente. El 80% de los encuestados prefirió las WLAN.



Preferencias sobre el uso de LAN's Inalámbricas

Uso de WLAN en la Actualidad

El uso más frecuente de las WLAN es como extensión de una red cableada de modo que se brinde una conexión a un usuario final móvil, por ejemplo:

- En hospitales, los datos del paciente pueden ser transmitidos de forma instantánea.
- En pequeños grupos de trabajo que necesiten una puesta en marcha rápida de una red (por ejemplo, grupos de revisión de estados de cuentas).
- En entornos dinámicos se minimiza la sobrecarga causada por extensiones de redes cableadas, movimientos de éstas u otros cambios instalando red sin cable.
- En centros de formación, universidades, corporaciones, etc., donde se usa red sin cable para tener fácil acceso a la información.
- En viejos edificios que se deban conservar en buen estado por aspectos estructurales, legales o históricos.
- Los trabajadores de almacenes pueden intercambiar información con una base de datos central mediante red sin cables, aumentando la productividad.

Tecnologías WLAN

Según el diseño requerido, se tienen distintas tecnologías WLAN aplicables:

- *Banda estrecha*. Se transmite y recibe en una banda de frecuencia específica, lo más estrecha posible para el paso de información. Los usuarios tienen distintas frecuencias de comunicación de modo que se evitan las interferencias. Así mismo, un filtro en el receptor de radio de cada terminal se encarga de dejar pasar únicamente la señal esperada en la frecuencia asignada.
- *Banda ancha*. Es el usado por la mayor parte de los sistemas sin cable. Fue desarrollado con fines militares para una comunicación segura, fiable y en misiones críticas. Se consume más ancho de banda para la transmisión, pero la señal es más fácil de detectar. El receptor conoce los parámetros de la señal que se ha difundido. En caso de que el receptor no esté en la frecuencia correcta, la señal aparece como ruido de fondo. Existen dos tipos de tecnología en banda ancha:
 - o a) *Frecuencia esperada de banda ancha (FHSS: Frequency-Hopping Spread Spectrum)*. Utiliza una portadora de banda estrecha que cambia su frecuencia periódicamente en un patrón conocido por transmisor y receptor. Convenientemente sincronizado, es como tener un único canal lógico. Para un receptor no sincronizado FHSS es como un ruido de impulsos de corta duración.

- o b) *Secuencia directa de banda ancha (DSSS: Direct-Sequence Spread Spectrum)*. Se genera uno o más bits redundantes por cada bit transmitido. Cuanto mayor sea esta secuencia de bits, mayor es la probabilidad de reconstruir los datos originales (aunque también se requiere mayor ancho de banda). Incluso si uno o más bits son perturbados en la transmisión, las técnicas implementadas en el receptor pueden reconstruir los datos originales sin necesidad de retransmitir. Para un receptor cualquiera, DSSS es un ruido de baja potencia.

Configuraciones de las WLAN

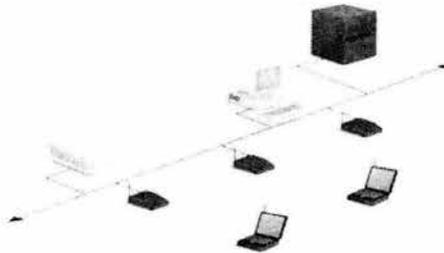
Las WLAN pueden ser simples o complejas. La configuración más básica se da cuando tenemos dos equipos con tarjetas adaptadoras para WLAN, de modo que pueden poner en funcionamiento una red independiente siempre que estén dentro del área de cobertura. Esto es llamado red uno a uno (peer to peer). Existen configuraciones más complejas, y a continuación se describirán las más comunes.

Red PPP (peer-to-peer). Cada cliente tendría únicamente acceso a los recursos de otro cliente pero no a un servidor central. Este tipo de redes no requiere administración o preconfiguración.



Red peer-to-peer

Cliente y Punto de Acceso. Instalando Puntos de Acceso (Access Points, AP's) se puede extender el rango al cuál los dispositivos pueden comunicarse, pues actúan como repetidores. Cualquier cliente tiene acceso a los recursos de la red cableada. Cada punto de acceso puede servir a varios clientes, según la naturaleza y número de transmisiones que tienen lugar. Existen muchas aplicaciones en el mundo real, conectando desde 15 y hasta 50 dispositivos clientes en un solo punto de acceso.



Cliente y punto de acceso

Roaming con Múltiple punto de Acceso. Los puntos de acceso tienen un rango finito, del orden de 150m en lugares cerrados y 300m en zonas abiertas. En zonas grandes, como por ejemplo un campus universitario o un edificio, es probablemente necesario más de un punto de acceso. La meta es cubrir el área total con células que solapen sus propias áreas, de modo que los clientes puedan moverse (sin cortes de señal) entre áreas de cobertura de diferentes puntos de acceso. Esto es llamado "roaming".



Múltiples puntos de acceso y "roaming"

Puntos de Extensión. Para resolver problemas particulares de topología, el diseñador de la red puede elegir usar un Punto de Extensión (Extension Points, EP's) para aumentar el número de puntos de acceso a la red. Los EP's funcionan como puntos de acceso, pero no están conectados directamente a la red cableada. Los puntos de extensión sirven para extender el rango de la red, retransmitiendo las señales desde un cliente a un punto de acceso o a otro punto de extensión. Los puntos de extensión pueden encadenarse para pasar mensajes a otros puntos de acceso, puntos de extensión o clientes más lejanos, de modo que se construye un "puente" entre ambos.



Uso de un punto de extensión.

Antenas Direccionales. Uno de los últimos componentes a considerar en el equipo de una WLAN es la antena direccional. Por ejemplo, si se quiere una LAN sin cable a otro edificio a 1Km de distancia. Una solución puede ser instalar una antena en cada edificio con línea de visión directa. La antena del primer edificio está conectada a la red cableada mediante un punto de acceso. Igualmente en el segundo edificio se conecta un punto de acceso, lo cuál permite una conexión sin cables entre ambos edificios.



Utilización de antenas direccionales.

La flexibilidad y movilidad de las redes inalámbricas las hace muy efectivas para extender una red, y son una atractiva alternativa a las redes cableadas, puesto que proporcionan la misma funcionalidad pero sin las restricciones físicas del cable.

Las redes inalámbricas permiten desde topologías simples y hasta las más complejas que ofrecen conexión, distribución de datos y "roaming". Además de ofrecer al usuario final movilidad en un entorno de red, habilitan redes portátiles.

Factores a considerar en una WLAN

Son varios los factores a considerar a la hora de comprar un sistema inalámbrico para la instalación de una red LAN. Algunos de los aspectos a tener en cuenta son los siguientes.

Cobertura

La distancia que pueden alcanzar las ondas de Radiofrecuencia (RF) o de Infrarrojos (IR) es función del diseño del producto y del ambiente de propagación, especialmente en lugares cerrados. Las interacciones con objetos, paredes, metales, e incluso la gente, afectan la propagación de la energía. Los objetos sólidos bloquean las señales de infrarrojos, lo cual impone límites adicionales. La mayoría de los sistemas de redes inalámbricas usan radio frecuencia porque pueden penetrar la mayor parte de obstáculos en lugares cerrados. El rango de cobertura de una LAN inalámbrica típica va de 10m. a los 100m. Puede extenderse y tener un alto grado de libertad y movilidad utilizando puntos de acceso que permiten el "roaming".

Rendimiento

El rendimiento de una LAN, depende de varios factores: de la configuración de los productos, del número de usuarios de la red, de los factores de propagación (cobertura, ambientes de propagación) y del tipo de sistema inalámbrico utilizado. Igualmente depende del retardo y de los cuellos de botella de la parte cableada de la red. Los usuarios de Ethernet o Token Ring no deberían experimentar gran diferencia en el rendimiento cuando utilizan una red inalámbrica. Estas proporcionan suficiente velocidad para las aplicaciones más comunes de una LAN en un puesto de trabajo, incluyendo correo electrónico, acceso a periféricos compartidos, acceso a Internet, acceso a bases de datos y a aplicaciones multiusuario.

Integridad y Fiabilidad

Estas tecnologías para redes inalámbricas se han probado durante más de 50 años en sistemas comerciales y militares. Aunque las interferencias de radio pueden degradar el rendimiento, ocurren raramente en el lugar de trabajo. Los robustos diseños de las ya probadas tecnologías para LAN inalámbricas y la limitada distancia que recorren las señales, proporciona conexiones que son mucho más robustas que las conexiones de teléfonos móviles y proporcionan integridad de datos de igual o mejor manera que una red cableada.

Compatibilidad con Redes Existentes

La mayor parte de LAN's inalámbricas proporcionan un estándar de interconexión con redes cableadas como Ethernet o Token Ring. Los nodos de la red inalámbrica son soportados por el sistema de la red de la misma manera que cualquier otro nodo de una red LAN, aunque se requieren de los controladores apropiados. Una vez instalado, la red trata los nodos inalámbricos igual que cualquier otro componente de la red.

Interoperabilidad

Los consumidores deben ser conscientes de que los sistemas inalámbricos de redes LAN de distintos vendedores pueden no ser compatibles para operar juntos. Existen tres razones principales:

- Uso de diferentes tecnologías. Un sistema basado en la tecnología de Frecuencia esperada (FHSS), no se comunicará con otro basado en la tecnología de Secuencia directa (DSSS).
- Sistemas que utilizan distinta banda de frecuencias no podrán comunicarse aunque utilicen la misma tecnología.
- Aún utilizando productos de igual tecnología y banda de frecuencias, los sistemas de diferentes vendedores pueden no comunicarse debido a diferencias de implementación de cada fabricante.

Interferencia y Coexistencia

La naturaleza en que se basan las redes inalámbricas implica que cualquier otro producto que transmita energía a la misma frecuencia, puede potencialmente ocasionar cierto grado de interferencia en un sistema LAN inalámbrico.

Pongamos como ejemplo a los hornos de microondas, aunque la mayor parte de fabricantes en la actualidad diseñan sus productos teniendo en cuenta las interferencias por microondas. Otro problema es la colocación de varias redes inalámbricas en lugares próximos. Mientras unas redes inalámbricas de unos fabricantes interfieren con otras redes, hay otras que coexisten sin interferencia. Este asunto debe tratarse directamente con los vendedores del producto.

Licencias

En los Estados Unidos, La Comisión Federal de Comunicaciones (FCC), gobierna la radio-transmisión, incluida la empleada en las redes inalámbricas. Otros países tienen sus correspondientes organismos reguladores (en México es la COFETEL). Típicamente las redes inalámbricas se diseñan para operar en porciones del espectro de radio donde el usuario final no necesita una licencia para utilizar las ondas de radio. En los Estados Unidos, la mayor parte de las redes difunden en una de las bandas de ISM (de instrumentación, científicas o médicas). Estas incluyen 902-928MHz, 2.4-2.483GHz, 5.15-5.35GHz, y 5.725-5.875GHz. Para poder vender productos de sistemas de LAN inalámbricos en un país en particular, el fabricante debe asegurar la certificación por el organismo encargado en ese país.

Simplicidad y Facilidad de Uso

Los usuarios necesitan adquirir muy pocos conocimientos además de lo que ya saben sobre redes LAN en general, para utilizar una LAN inalámbrica. Esto es así porque la naturaleza inalámbrica de la red es transparente al usuario, las aplicaciones trabajan de igual manera que lo hacían en una red cableada, los productos de una LAN inalámbrica incorporan herramientas de diagnóstico para dirigir los problemas asociados a los elementos inalámbricos del sistema, de forma que los usuarios rara vez tengan que utilizarlos. Las LAN inalámbricas simplifican muchos de los problemas de instalación y configuración que atormentan a los que dirigen la red. Ya que únicamente los puntos de acceso de las redes inalámbricas necesitan cable, ya no es necesario llevar cable hasta el usuario final. La falta de cable hace también que los cambios, extensiones y desplazamientos sean operaciones triviales en una red inalámbrica. Finalmente, la naturaleza portable de las redes inalámbricas permite a los encargados de la red preconfigurar ésta y resolver problemas antes de su instalación en un lugar remoto. Una vez configurada, la red puede llevarse de un lugar a otro con muy poca o ninguna modificación.

Seguridad en la Comunicación

Puesto que la tecnología inalámbrica se ha desarrollado en aplicaciones militares, la seguridad ha sido uno de los criterios de diseño para los dispositivos inalámbricos.

Normalmente se suministran elementos de seguridad dentro de la LAN inalámbrica, haciendo que estas sean más seguras que la mayor parte de redes cableadas. Es muy complicado que los receptores no sintonizados escuchen el tráfico que se da en la LAN. Complejas técnicas de cifrado hacen prácticamente imposible acceder de forma no autorizada al tráfico de la red. En general los nodos individuales deben tener habilitada la seguridad antes de poder participar en el tráfico de la red.

Costo

La instalación de una LAN inalámbrica incluye los costos de infraestructura para los puntos de acceso y los costos de usuario para los adaptadores de la red inalámbrica. Los costos de infraestructura dependen fundamentalmente del número de puntos de acceso desplegados. El valor de los puntos de acceso oscila entre 1,000 y 2,000 dólares. El número de puntos de acceso depende de la cobertura requerida y del número y tipo de usuarios. El área de cobertura es proporcional al cuadrado del rango de alcance de los productos adquiridos. Los adaptadores son requeridos para las plataformas estándar de computadoras y su precio oscila entre 300 y 1,000 dólares. El costo de instalación y mantenimiento de una WLAN generalmente es más bajo que el costo de instalación y mantenimiento de una red cableada tradicional, por dos razones:

En primer lugar una red WLAN elimina directamente los costos de cableado y el trabajo asociado con la instalación y reparación. En segundo lugar una red WLAN simplifica los cambios, desplazamientos y extensiones, por lo que se reducen los costos indirectos de los usuarios.

Escalabilidad

Las redes WLAN pueden ser diseñadas para ser extremadamente simples o bastante complejas. Las WLAN's pueden soportar un amplio número de nodos y/o extensas áreas físicas añadiendo puntos de acceso para dar energía a la señal o para extender la cobertura.

Seguridad Laboral

La potencia de salida de los sistemas WLAN es muy baja, mucho menor que la de un teléfono móvil. Puesto que las señales de radio se atenúan rápidamente con la distancia, la exposición a la energía de radio-frecuencia en el área de la WLAN es muy pequeña. Las WLAN's deben cumplir las estrictas normas de seguridad dictadas por el gobierno y la industria. No se han atribuido nunca efectos secundarios en la salud a causa de una WLAN.

3.4.2. Estándares de Comunicaciones Inalámbricas

Telefonía Celular

En lo que a telefonía celular respecta, actualmente existen una gran variedad de estándares de transmisión. Esta situación no unificada impide, entre otras cosas, que existan servicios como el *roaming* mundial. Algunos de los estándares más importantes de la actualidad son:

CDMA (code division multiple access, acceso múltiple por división de códigos)

Esta tecnología posee una capacidad 10 veces mayor a la de la telefonía analógica y permite utilizar el espectro con más eficiencia. Para ello, el sistema asigna un código único a cada transmisor, de manera que muchas llamadas pueden compartir una misma frecuencia, y las distintas llamadas se diferencian entre sí gracias a sus respectivos códigos. Además, esa característica le da más seguridad a las llamadas, pues la codificación las protege de oídos ajenos. Las compañías más "casadas" con este estándar son Qualcomm (el inventor de la misma) y Motorola.

TDMA (time division multiple access, acceso múltiple por división de tiempo)

También utilizada para separar múltiples conversaciones sobre una frecuencia en el ancho de banda, esta tecnología divide las llamadas asignando a cada usuario una ranura de tiempo específica. Un sistema celular digital que utiliza el estándar TDMA asigna 10 espacios de tiempo a cada canal de la frecuencia; los teléfonos celulares dividen su señal en paquetes o información que mandan separados, durante esos intervalos, y los paquetes son reensamblados por el equipo que los recibe. Una de las compañías que da más apoyo a esta tecnología es Ericsson.

GSM (group special mobile, ahora conocido como global system for mobile communications, sistema global para comunicaciones móviles)

Este estándar es el que se utiliza en redes celulares digitales en Europa, Australia y Japón. Su principal virtud es la forma en que permite la interoperabilidad entre redes y países, con lo cual se logra *roaming* internacional para los usuarios. Opera en la banda de los 900MHz. Nokia y Ericsson son dos proveedores de infraestructura muy comprometidos con este estándar.

WCDMA (Wideband CDMA, CDMA de banda ancha)

Es la versión futura de CDMA. Optimizada para permitir servicios como voz, datos y video, permite un ancho de banda de 2Mbps en el área local, y de 384kbps en un área extensa con total movilidad. Debido a que transmite datos a tasas elevadas, requiere una banda ancha de radiofrecuencia, y es por ello que funciona con una portadora de 5MHz, en contraste con los 200kHz que GSM requiere por ejemplo.

3G (Tercera Generación)

Uno de los aspectos que se buscan para la Tercera Generación (3G) es la adopción de un estándar único que pueda ser adoptado por todas las compañías que fabrican redes y teléfonos celulares, así como por parte de los operadores.

Análisis

Sin embargo, muchos de los involucrados tienen intereses y preferencias a favor de una u otra tecnología, lo cual ha dificultado enormemente la ruta hacia una plataforma unificada. Algunas de las capacidades que se espera que 3G brinde son:

- Multimedia mejorada (voz, datos, video y control remoto).
- Capacidad de operar en los modos más populares (telefonía celular, e-mail, localizadores, fax, videoconferencias y navegación Web).
- Banda ancha y alta velocidad (hasta 2Mbps).
- Flexibilidad de ruteo (repetidores, satélites, LAN).
- Operación de frecuencias de 2GHz de envío y recepción.
- Capacidad de roaming en Europa Japón y Norteamérica.

Transmisión de Datos

En lo que se refiere a transmisión de datos para sistemas de cómputo los principales estándares para redes inalámbricas son:

IrDA

IrDA (Infrared Data Association) es una organización establecida en 1993 patrocinada por la industria, con el fin de crear estándares internacionales para el hardware y software que se emplea en esquemas de comunicación por luz infrarroja. En esta especial forma de radio transmisión, un rayo enfocado de luz en el espectro de frecuencia del infrarrojo (del orden de los terahertz, o billones de hertz) es modulado con información y enviado desde un transmisor a un receptor a una distancia relativamente corta. La radiación infrarroja (IR) es la misma tecnología empleada para manejar un televisor a control remoto. La comunicación de datos vía infrarrojo está jugando un papel importante en la comunicación inalámbrica de datos debido a la popularidad de computadoras portátiles, asistentes digitales personales (PDA's), cámaras digitales, teléfonos móviles, buscadores personales y otros dispositivos. Entre los usos actuales están:

- Envío de documentos desde una computadora portátil a una impresora.
- Intercambiar tarjetas de negocios entre computadoras de bolsillo.
- Coordinar agenda y directorios telefónicos entre computadoras de escritorio y portátiles.
- Enviar faxes desde una computadora portátil hacia una máquina de fax distante a través de un teléfono público.
- Cámaras digitales que pueden enviar imágenes a una computadora.

La comunicación a través de infrarrojo involucra un "transceiver" (receptor y emisor) en ambos dispositivos que se comunican. Microcircuitos especiales proveen esta capacidad. En el estándar IrDA-1.1, el máximo tamaño de información que puede ser transmitido es 2048bytes y la máxima tasa de transmisión es de 4Mbps.



Bluetooth

La tecnología Bluetooth es una especificación abierta para la comunicación inalámbrica de datos y voz. Está basada en un enlace de radio de bajo costo y corto alcance, proporcionando conexiones instantáneas para entornos de comunicaciones tanto móviles como estáticas. En definitiva, Bluetooth pretende ser una especificación global para la conectividad inalámbrica.

El principal objetivo de esta tecnología, es la posibilidad de reemplazar los muchos cables propietarios que conectan unos dispositivos con otros, por un medio de enlace de radio universal de corto alcance. Por ejemplo, la tecnología de radio Bluetooth implementada en el teléfono celular y en el dispositivo portátil reemplazaría el molesto cable utilizado hoy en día para conectar ambos aparatos. Las impresoras, las agendas electrónicas, los PDA, los faxes, los teclados, los joysticks y prácticamente cualquier otro dispositivo digital son susceptibles de formar parte de un sistema Bluetooth.



Pero más allá de reemplazar los incómodos cables por un enlace inalámbrico, la tecnología Bluetooth ofrece un puente a las redes de datos existentes, una interfaz con el exterior y un mecanismo para formar, en el momento, pequeños grupos de dispositivos conectados entre sí de forma privada fuera de cualquier estructura fija de red.

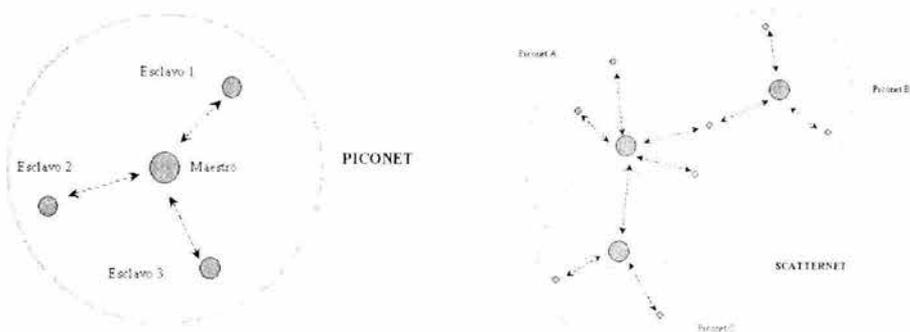
Análisis

Integrado en un pequeño transmisor de radiofrecuencia, Bluetooth permite conectar entre sí todo tipo de dispositivos electrónicos (teléfonos, computadoras, impresoras, faxes, etc.) situados dentro de un radio limitado a 10 metros (ampliable a 100 metros, aunque con mayor distorsión) sin necesidad de utilizar cables. El transmisor está integrado en un pequeño circuito de 9x9 milímetros y opera en una frecuencia de banda global (2,4GHz, utilizada en muchos países para usos médicos y científicos) que asegura la compatibilidad universal. Los dispositivos que incorporan Bluetooth se reconocen y se hablan de la misma forma que lo hace, por ejemplo, una computadora con su impresora.

El canal permanece abierto y no requiere la intervención directa y constante del usuario cada vez que se quieren enviar datos. El transmisor permite enviar voz y datos a una velocidad máxima de 700kbps. y consume un 97% menos energía que un teléfono móvil. Además, es inteligente: cuando el tráfico de datos disminuye el transmisor adopta el modo de bajo consumo de energía. Las diferentes partes del sistema Bluetooth son:

- Una unidad de radio.
- Una unidad de control del enlace.
- Dispositivo de gestión del enlace.
- Funciones de software.

El sistema Bluetooth permite conexiones punto a punto y punto a multipunto. La velocidad de datos teórica en full-dúplex dentro de una red como la ilustrada a continuación es de 6Mbps. Una piconet es la colección de dispositivos (de 2 a 8 máximo) conectados por medio de la tecnología Bluetooth. Todos los dispositivos tienen la misma implementación del protocolo; sin embargo, al crearse la red, una unidad actuará como maestra y el resto como esclavas mientras dure la conexión. Múltiples *piconets* que comparten dispositivos entre sí, pueden formar en conjunto una red de *piconets*. Esto se denomina *scatternet*.



Redes Bluetooth

Wi-Fi

Wi-Fi o *wireless fidelity* es un término promulgado por la Wi-Fi Alliance y se emplea de manera genérica cuando se refiere a cualquier tipo de red que cumpla con el estándar IEEE 802.11 y/o cualquiera de sus variantes. Todos los productos que hayan sido probados y aprobados como "Wi-Fi Certified"^{*} por la Wi-Fi Alliance, están certificados como interoperables entre sí, incluso si son de diferentes fabricantes. Un usuario puede usar cualquier punto de acceso de cualquier marca de hardware con cualquier cliente que también esté certificado. Sin embargo, esto no se garantiza para equipos que operen a diferente frecuencia de radio.

802.11 es el estándar del IEEE (Institute of Electrical and Electronics Engineers) para redes inalámbricas. Una red 802.11 se comunica enviando paquetes de datos Ethernet a través del aire. El estándar permite la integración inalámbrica con cableado IEEE 802.3 (red Ethernet) usando puntos de acceso o de extensión. Esto significa que el estándar inalámbrico IEEE 802.11 soporta todos los estándares de protocolos de red Ethernet incluyendo TCP/IP, AppleTalk, IPX y NetBEUI.

El estándar inalámbrico 802.11 actualmente incluye 3 tipos diferentes de tecnologías de radio:

- Frequency Hopped Spread Spectrum (FHSS).
- Direct Sequence Spread Spectrum (DSSS).
- Orthogonal Frequency Division Multiplexing (OFDM).

FHSS es una tecnología de transmisión empleada en transmisiones WLAN donde la señal de datos es modulada con un portador de banda estrecha que "salta" de frecuencia en frecuencia de manera aleatoria pero en una secuencia predecible como una función del tiempo en una banda amplia de frecuencias. La energía de la señal es propagada en el dominio del tiempo en lugar de cortar cada bit en pequeñas piezas en el dominio de la frecuencia.

Esta técnica reduce la interferencia debido a que una señal desde un sistema de banda estrecha sólo afectará la señal si ambas son transmitidas en la misma frecuencia y al mismo tiempo. Si se sincronizan apropiadamente, un canal lógico único es mantenido. Las frecuencias de transmisión son determinadas por un código de extensión o salto. El receptor debe estar configurado con el mismo código de salto y debe escuchar a la señal entrante en el momento preciso y con la frecuencia correcta de manera que reciba apropiadamente la señal.

Las regulaciones actuales de la FCC requiere que los fabricantes usen 75 o más frecuencias por canal de transmisión con un máximo tiempo de espera (tiempo que se espera en una frecuencia en particular antes de cualquier salto) de 400ms.

DSSS usa un canal fijo para comunicaciones. Un dispositivo inalámbrico *DSSS* necesita estar en el mismo canal que el Punto de Acceso para estar apto para comunicarse con él.

^{*} Marca Registrada por la Wi-Fi Alliance

Esta técnica ofrece un mayor rendimiento porque se usan canales fijos, y se puede mejorar configurando múltiples puntos de Acceso en diferentes canales. De esta manera los usuarios no competirán por el mismo ancho de banda todo el tiempo a través de un único Punto de acceso. DSSS 802.11 trabaja en la Banda de Frecuencia 2.4GHz la cuál no requiere una licencia en la mayoría de los países y provee alto rendimiento de bajas frecuencias. Catorce canales de 2.4GHz son permitidos para uso a nivel mundial; 1-11 en USA y Canadá, 1-13 en Europa, 12-13 en España, 10-13 en Francia, y 1-14 en Japón.

DSSS genera un patrón de bit redundante por cada bit a ser transmitido. Este bit patrón es llamado un chip. Una mayor longitud del chip, ofrece una probabilidad mayor de que los datos puedan ser recuperados. Si uno o más bits en el chip son 'dañados' durante la transmisión, se pueden recuperar los datos originales a través de técnicas estadísticas aplicadas sobre las señales de radio, sin necesidad de retransmisiones. La mayoría de los fabricantes de productos para redes inalámbricas, han adoptado la tecnología DSSS después de considerar los beneficios, comparándolos con los costos y rendimiento que se obtienen con ella.

OFDM es una tecnología de multiplexado que transmite múltiples señales simultáneamente sobre una ruta de transmisión, tales como un sistema de cable o inalámbrico. Cada señal viaja en su propio y único rango de frecuencias (portador), el cual es modulado por los datos (texto, voz, video, etc.). La técnica FDM Ortogonal de propagación de espectro distribuye los datos sobre un gran número de portadores que están separados en frecuencias precisas. Este espaciamiento provee la "ortogonalidad" de esta técnica, la cual previene que los demoduladores vean otras frecuencias que no sean las suyas. Los beneficios de OFDM son una alta eficiencia espectral, mayor inmunidad a interferencia de RF y baja distorsión. Debido a que múltiples versiones de la señal interfieren una con la otra (Inter Symbol Interference , ISI) se vuelve muy difícil el extraer la información original.

Familia 802.11

Existen varias especificaciones dentro de la familia 802.11, las cuales se describen a continuación:

- *802.11*. Empleada para LAN's inalámbricas. Provee velocidades de transmisión de 1 a 2Mbps en la banda de 2.4GHz usando FHSS (frequency hopping spread spectrum) o DSSS (direct sequence spread spectrum).
- *802.11a*. Es una extensión a 802.11 que aplica a LAN's inalámbricas y provee velocidades de transmisión de hasta 54Mbps en la banda de 5GHz con 12 canales separados no traslapables. Como resultado de esto, se pueden tener 12 Puntos de Acceso configurando cada uno en un distinto canal en la misma área sin que interfiera uno con el otro, lo que hace más fácil la configuración y asignación de canal e incrementa el desempeño de la red. 802.11a usa únicamente codificación OFDM. La mayor frecuencia permite que el alcance sea de aproximadamente 30 a 40 metros, se tenga más facilidad para atravesar muros y obstáculos y se mantenga una velocidad constante a lo largo del rango. La señal es menos propensa a interferencias por estar en una banda poco poblada. No tiene compatibilidad con estándares de 2.4GHz.

- *802.11b*. También conocido como 802.11 de alta velocidad, es una extensión a 802.11 que se emplea en LAN's inalámbricas y provee velocidades de transmisión de 11Mbps (con soporte para 5.5, 2 y 1Mbps) en la banda de 2.4GHz. 802.11b emplea únicamente DSSS y permite una funcionalidad comparable a Ethernet
- *802.11g*. Es una extensión al estándar 802.11b que amplía el ancho de banda a los 54Mbps dentro de la banda de los 2.4GHz empleando tecnología OFDM (orthogonal frequency division multiplexing). Tiene compatibilidad con protocolos anteriores como 802.11b, aunque pierde la ventaja de velocidad cayendo a 11Mbps o menores. Cualquier tarjeta 802.11b se comunicará con Puntos de Acceso 802.11g y bastará una actualización de firmware para actualizar un 802.11b a 802.11g. Se tienen sólo 3 canales no traslapados, de manera que se tiene la misma dificultad que 802.11b en la asignación de canales cuando se quiere cubrir una extensa zona y un gran número de usuarios. Tiene un alcance de 100 m., aunque el desempeño de la red cae mucho conforme se aproxima al rango máximo. Debido a que opera en la banda de 2.4GHz es más propensa a interferencias de otros equipos electromecánicos, como son los teléfonos inalámbricos.

Wireless USB (WUSB)

En febrero del 2004, se crea oficialmente un consorcio de empresas líderes en comunicaciones conformado, entre otros, por Intel, Agere Systems, Hewlett Packard, Microsoft Corp., NEC, Philips Semiconductors y SamsungElectronics. Este consorcio es bautizado con el nombre de "Wireless USB Promoter Group" y su misión es definir y promover un estándar de comunicación denominado Wireless USB (USB inalámbrico).

Básicamente, la tecnología WUSB, brindará exactamente la misma funcionalidad y trabajará con la misma topología que el estándar USB con cables. Estará basado en el trabajo *Ultra-WideBand* (UWB) de la *MultiBand OFDM Alliance* y la *WiMedia Alliance*.

WUSB cumplirá con el estándar USB 2.0, y tendrá un alcance similar al de bluetooth de 10 metros. Sin embargo, y a diferencia de bluetooth, será un medio de comunicación de mediano consumo de energía debido que dispondrá de un ancho de banda igual al de USB (480Mbps), por lo que estará orientado a dispositivos un poco más potentes, como laptops o PDA's de mayor poder.

La primera especificación completa y terminada de WUSB se espera para finales del 2004.

3.4.3. Java Inalámbrico.

Introducción a la tecnología inalámbrica Java

Para desarrollar aplicaciones empleando tecnología Java, se necesita asimilar mucha información de diversos campos. Se requiere entender un poco sobre tecnologías de comunicaciones inalámbricas, el negocio de las comunicaciones inalámbricas y mucho sobre la plataforma Java. Aquí se ofrece un vistazo sobre la tecnología inalámbrica Java y las tecnologías que integra.

Comunicaciones Inalámbricas

Las comunicaciones inalámbricas son un campo enorme, abarcando desde radio y televisión, pasando por localizadores y teléfonos móviles, hasta las comunicaciones satelitales. El campo de los teléfonos móviles se está expandiendo muy rápido, al mismo tiempo que los estándares y protocolos están siendo adoptados, usados, actualizados y algunas veces descartados. La otra parte del mundo inalámbrico que está expandiéndose rápidamente es el de las redes de área local inalámbricas (WLAN's). Controlado por un estándar de amplia aceptación, el IEEE 802.11, las redes locales inalámbricas para computadoras están creciendo y siendo aceptadas de una manera vertiginosa.

Aunque red inalámbrica pareciera ser sólo un caso especial de una red, es de hecho más intuitivo y más natural que las redes alambradas. Algún día, probablemente muy pronto, la necesidad de conectar físicamente un dispositivo en una red parecerá pintoresco y anticuado. La noción de que uno pudiera caminar en una habitación con un teléfono celular y no le fuera posible el interactuar con otros dispositivos en esa habitación parecerá increíblemente primitivo. El futuro revelará que las redes alambradas son de hecho los casos especiales. Conceptualmente, las comunicaciones inalámbricas pueden dividirse en dos tipos, de área local y de área amplia. Un dispositivo local es similar al transmisor llavero que con un botón abre las puertas de un auto, un teléfono inalámbrico de 900MHz, un juguete de radio control o incluso las nuevas redes Bluetooth. Todos estos dispositivos operan en rangos de distancias muy pequeños, usualmente de sólo unos pocos metros.

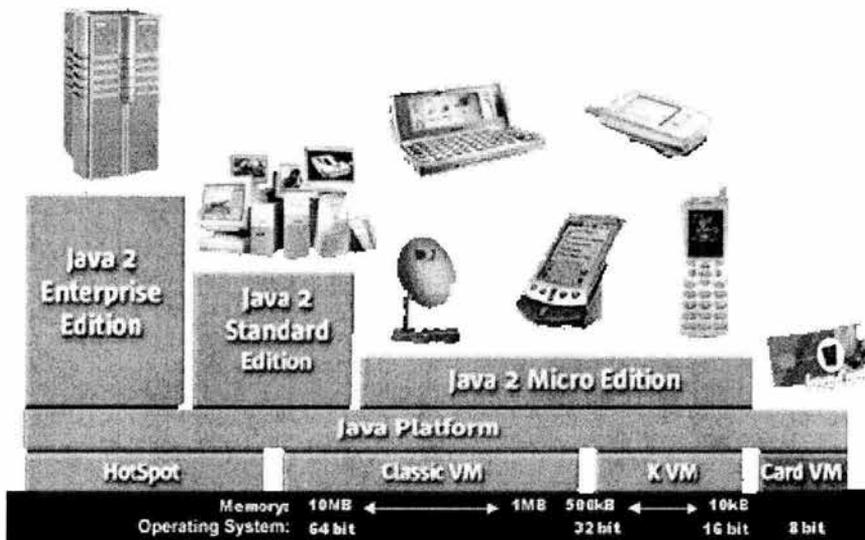
Dispositivos inalámbricos de área amplia operan eficientemente sobre un área mucho mayor. Los rangos mayores de estos dispositivos, sin embargo, están basados en un truco: una red alambrada más elaborada. En realidad un celular no tiene mucho más poder de transmisión que un juguete de radio control. Lo que se hace, es tener una red de antenas de radio (llamadas células) cuidadosamente posicionadas; el teléfono puede seguir operando mientras se encuentre dentro del rango de, al menos, una célula de la red. Así, este teléfono celular recibe servicio de un portador inalámbrico: una compañía que opera dicha red.

Las tecnologías inalámbricas emergentes están abriendo un mercado completamente nuevo para nuevos estilos de aplicaciones y servicios orientados a consumidores y empresas. La tecnología Java provee los fundamentos que permiten a los dispositivos de nueva generación ofrecer nuevas capacidades tales como una mejor interactividad, ricas interfaces de usuario, procesamiento fuera de línea, almacenamiento de datos local, y entorno en red. Al utilizar estas nuevas capacidades, los desarrolladores y empresas pueden crear excitantes nuevos servicios en el mercado inalámbrico.

A continuación se provee una descripción de los beneficios provistos por la tecnología Java en un mundo inalámbrico. Gracias a las características de Java, se superan los retos presentados por la gran variedad de teléfonos móviles, PDA's, dispositivos inalámbricos y tecnologías de red, permitiendo construir aplicaciones seguras y fáciles de usar.

La plataforma Java en un mundo Wireless

Java 2 Micro Edition (J2ME), forma parte de la plataforma Java 2. Mientras que Java 2 Standard Edition (J2SE) se enfoca en sistemas de escritorio, y Java 2 Enterprise Edition (J2EE) se enfoca en las aplicaciones empresariales para servidores, J2ME es una colección de API's destinadas a dispositivos para el consumidor final y sistemas inmersos, que van desde codificadores de TV y sistemas de navegación a bordo, hasta teléfonos móviles y PDA's. Con cada edición de la plataforma Java 2, hay diferentes Máquinas Virtuales de Java (JVM) que son implementaciones ya optimizadas para el tipo de sistemas a los cuales se enfocan. Por ejemplo, la K Virtual Machine (KVM) es una JVM optimizada para dispositivos con pocos recursos, tales como teléfonos celulares y PDA's.



J2EE es parte de la Plataforma Java 2

Las siguientes características están compartidas entre las 3 ediciones de Java:

- Write Once Run Anywhere: Posibilidad de escribir y compilar una sola vez un programa en Java y correrlo sin ningún problema en cualquier máquina con cualquier plataforma y cualquier sistema operativo compatibles con Java.
- Seguridad: La tecnología Java brinda un robusto modelo de seguridad, por medio de su preverificador de código, administradores de seguridad, el modelo de sandbox, uso de SSL, etc.

- Interfaz de usuario: Los dispositivos móviles se pueden beneficiar de API's de interfaces de usuario más ricas, que permiten una mejor interacción con el usuario.
- Conocimiento de la red: Mientras que las aplicaciones Java pueden operar en modo desconectado, éstas siempre tienen conocimiento de la red por defecto, permitiendo que las aplicaciones sean dinámicamente descargadas. Adicionalmente, Java es indiferente de la red, en el sentido de que las aplicaciones pueden intercambiar información con un servidor sobre cualquier protocolo de red (TCP/IP, WAP, GSM, CDMA, etc.).

El ciclo de las aplicaciones J2ME

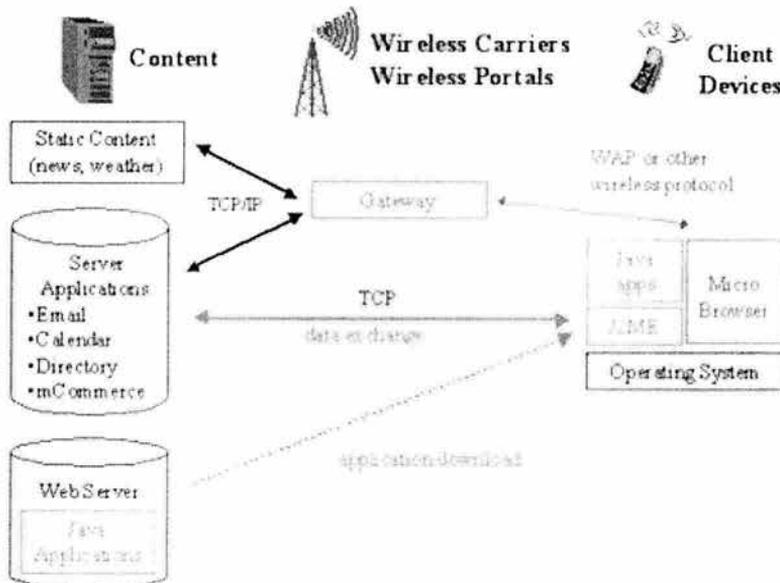
Contrario al modelo del Navegador Web, el cuál requiere conectividad continua y ofrece limitadas interfaces de usuario y seguridad, J2ME permite a las aplicaciones ser descargadas dinámicamente a un dispositivo móvil de una manera segura. Las aplicaciones J2ME pueden ser publicadas en un Servidor Web, permitiendo a usuarios finales el iniciar la descarga de una aplicación que seleccionen a través de un micro navegador o alguna otra interfaz. Operadores inalámbricos, proveedores de contenido e ISP's pueden producir un conjunto de aplicaciones J2ME y administrarlas remotamente. El modelo de funcionamiento de J2ME pone la responsabilidad de verificar la compatibilidad de las aplicaciones (tales como la versión de la especificación J2ME usada, librerías disponibles, hardware instalado y capacidades nativas proporcionadas por el dispositivo, etc.) en el dispositivo mismo, permitiendo al usuario final ignorar las contrariedades asociadas a los sistemas de escritorio típicos.

Una vez que la aplicación J2ME es desplegada en un dispositivo móvil, permanece ahí hasta que el usuario decide actualizarlo o removerlo. La aplicación puede ser operada en modo desconectado y almacenar información localmente. Debido a que la aplicación reside localmente, el usuario no experimenta ningún efecto de latencia, y la aplicación ofrece una interfaz de usuario (menús desplegables, cajas de diálogo, íconos animados, etc.). El nivel de conveniencia es incrementado por que el usuario puede controlar cuando una aplicación inicia el intercambio de datos sobre una red inalámbrica. Esto permite grandes ahorros en costos, en redes donde los usuarios son facturados por minuto, y permite un intercambio de datos más eficiente ya que muchas aplicaciones pueden usar un mecanismo de caché para minimizar la latencia en la red.

Redes Inalámbricas para Aplicaciones J2ME

Las Aplicaciones J2ME pueden intercambiar datos sobre virtualmente cualquier protocolo de red. Adicionalmente, las aplicaciones J2ME pueden aceptar cualquier infraestructura de red inalámbrica: se pueden comunicar a través de una red GSM, CDMA, TDMA, Bluetooth, Wi-Fi, etc. Las mismas aplicaciones están listas para ser usadas en redes basadas en paquetes, permitiendo el uso de protocolos estándares en Internet, tales como HTTPS (HTTP sobre SSL), IMAP o LDAP entre la aplicación cliente J2ME y la infraestructura del servidor.

A continuación se muestra una posible arquitectura de una red inalámbrica usando Java.



Red Inalámbrica con Java

Beneficios de J2ME en dispositivos inalámbricos

Demos una mirada al como la tecnología Java encaja en la evolución del servicio inalámbrico. Originalmente, la tecnología analoga era suficiente para manejar servicios de voz, pero la calidad de las llamadas era poca y múltiples redes de radio competían una con otra. Hoy día, tomamos ventaja de la segunda generación de redes y servicios (redes 2G), los cuales emplean redes digitales y tecnologías de navegadores de Web. Esto provee acceso a servicios de datos, pero los lenguajes de etiquetas presentan algunas limitaciones.

Los lenguajes de etiquetas presentan un paso en la dirección correcta, pero las aplicaciones basadas en Navegadores no trabajan cuando se encuentran fuera del espacio de cobertura incluso con operaciones simples (como entrar citas en un calendario basado en Navegador). Ofrecen un paradigma de interfaz de usuario limitado. Cuando la tecnología Java es agregada a este entorno, trae beneficios adicionales que se traducen en una experiencia de usuario mejorada. En lugar de aplicaciones de texto plano y latencia asociada a una interfaz basada en Navegador, al usuario se le presentan ricas gráficas animadas, una rápida interacción, la posibilidad de emplear una aplicación fuera de línea y quizá la más importante, la capacidad de descargar dinámicamente nuevas aplicaciones al dispositivo.

Para los desarrolladores de aplicaciones, esto significa que uno puede usar el lenguaje de programación y las herramientas de desarrollo preferidos, en lugar de aprender un nuevo ambiente de desarrollo.

Hay más de 2.5 millones de desarrolladores que ya han creado aplicaciones usando el lenguaje de programación Java, principalmente en el lado del servidor. Una vez que estos desarrolladores se familiaricen con el pequeño conjunto de API's J2ME, se vuelve relativamente fácil desarrollar pequeños módulos cliente que intercambien datos con aplicaciones de servidor sobre redes inalámbricas. Los retos que permanecen tanto para Java, WAP, o API's nativas, es que las pequeñas pantallas e interfaces de entrada limitadas requieren que los desarrolladores pongan mucho esfuerzo en el desarrollo de la interfaz de usuario de la aplicación. En otras palabras, los pequeños dispositivos obligan a los desarrolladores a abandonar técnicas de programación malas y/o perezosas.

¿Que tipo de Aplicaciones permite J2ME?

Mucha gente espera ver nuevos tipos de aplicaciones desarrollados con J2ME. Se puede anticipar que las categorías de las aplicaciones permanecerán igual, excepto por unas cuantas, como servicios de localización y aplicaciones de datos que se integran con funcionalidad telefónica. El resultado, muy probablemente, serán aplicaciones que son sensitivas al contexto (inmediación, ambiente, fecha, hora, uso personal o profesional) y que están migrando de una interfaz basada en caracteres (aplicaciones basadas en navegador) hacia un ambiente gráfico, proporcionando a los desarrolladores y usuarios finales un inigualable nivel de flexibilidad. Sólo pensemos sobre la evolución desde DOS o aplicaciones mainframe, hasta Windows, MacOS o el ambiente gráfico de Solaris. Aún usamos procesadores de palabras, hojas de cálculo, aplicaciones de contabilidad como en los buenos viejos tiempos, pero debido a que la nueva generación de aplicaciones toma ventaja de un ambiente gráfico más rico, las aplicaciones son mejores y más fáciles de usar.

Por consiguiente, esperemos ver desarrolladores creando las mismas categorías de aplicaciones que llegaron a existir para protocolos como WAP, pero en esta ocasión con la experiencia y el conocimiento necesarios para enriquecer la experiencia del usuario. Aún sin llegar a que J2ME sea adoptado, los pronósticos son realmente buenos. La consultora *Evans Data* recientemente realizó una encuesta entre 500 desarrolladores de aplicaciones inalámbricas, concluyendo que más desarrolladores usarán Java y J2ME para el desarrollo de aplicaciones (30%), que API's de C nativos (Palm OS, PocketPC, EPOC) o incluso WAP. Se estima que el mercado en el que J2ME penetrará de manera más rápida será el Japonés, estimado en una penetración de mercado del 40%. El principal proveedor Nipón de telefonía celular empezó a ofrecer teléfonos con soporte para J2ME en Enero del 2002 y había vendido más de 3 millones de unidades para finales de Septiembre del mismo año. Para finales del año, las otras dos más grandes operadores de servicios inalámbricos en Japón se unieron para desarrollar aplicaciones J2ME.

Los beneficios de la tecnología Java provistos por J2ME en el ámbito de redes inalámbricas son muchos y muy variados. Gracias a su flexibilidad ("write once, run anywhere"), su independencia de cualquier infraestructura inalámbrica, su soporte para procesamiento de datos fuera de línea, su almacenamiento de datos local y sus características de seguridad robustas, J2ME es la plataforma natural para el desarrollo de aplicaciones inalámbricas. No por nada el número de desarrolladores está creciendo rápidamente.

Alcance de La Tecnología Wireless Java

La tecnología Wireless Java es la intersección de dos vastos mundos, las comunicaciones inalámbricas de datos y la plataforma Java. Es la conjunción de partes de J2ME, Personal Java, J2SE y J2EE. Dicho esto, algunas confusiones sobre la tecnología inalámbrica de Java pueden aclararse. La tecnología Wireless Java y J2ME no son la misma cosa. J2ME es una plataforma que opera en dispositivos portátiles, que puede o no incluir Wireless Java; por otro lado, Wireless Java sirve para la comunicación inalámbrica entre dispositivos. La tecnología Wireless Java no está confinada únicamente a J2ME. Se podría tener una computadora o un PDA corriendo aplicaciones J2SE, conectándose a otras computadoras a través de una red 802.11 con Wireless Java. Para fines prácticos, consideremos una conexión Wireless como un punto de acceso que tiene un dispositivo a una red local (LAN). Otra confusión puede surgir al confundir MIDP con J2ME, pero cabe aclarar 2 cosas:

- MIDP no es todo de J2ME. MIDP es el primer perfil terminado y cuenta con la primera base terminada de dispositivos en el mundo, así que por ello la gente asume que se habla de J2ME cuando se habla de MIDP.
- MIDP no es todo de la tecnología Wireless Java. La plataforma Java ofrece muchas opciones para programación inalámbrica: Personal Java, J2SE en dispositivos inalámbricos e incluso los próximos perfiles para PDA's.

¿Por que Java para desarrollo de aplicaciones inalámbricas?

La plataforma Java es una excelente elección par el desarrollo de sistemas inalámbricos por muchas razones. Las tres obligadas son:

- La plataforma Java es segura. El código Java siempre se ejecuta dentro de los confines de la Máquina Virtual, lo cual provee un ambiente seguro para ejecutar código descargado. Una aplicación binaria podría ocasionar un error grave en un dispositivo (imaginemos una pantalla azul de error en nuestro teléfono móvil). En contraste, siendo pesimistas, una aplicación Java sólo podría inhabilitar la Máquina Virtual, y no todo el dispositivo.
- El lenguaje Java facilita y motiva la programación robusta. El "garbage colector" ahorra a los programadores incontables horas de programación y los permite despreocuparse del uso de la memoria. De igual forma, el mecanismo de excepciones les obliga a crear aplicaciones robustas.
- La portabilidad es una ganancia para la tecnología inalámbrica de Java. Un único ejecutable puede correr en muchos dispositivos. Por ejemplo, cuando se escribe un MIDlet (una aplicación MIDP), esté correrá en cualquier dispositivo que implemente la especificación MIDP. Un segundo beneficio de la portabilidad es la facilidad para liberar y distribuir las aplicaciones a diversos dispositivos a través de la red inalámbrica (algunas veces llamadas transmisión *on-the-air* o sobre-el-aire). Las aplicaciones binarias también pueden ser desplazadas de un servidor hacia un dispositivo, pero no de manera segura. Debido a que el código Java corre dentro de la JVM, el código es descargado de la red y se puede correr de manera segura. El código binario no puede ser controlado en tiempo de ejecución y es mucho menos seguro.

Mercado en las comunicaciones inalámbricas

Los proveedores de servicios, así como los desarrolladores de aplicaciones están actualmente experimentando con modelos de negocios que les permitan obtener ganancias en el recién creado, pero muy atractivo y creciente mercado de las comunicaciones inalámbricas. Las compañías proveedoras de servicios esperan brindar un valor agregado a sus aplicaciones (que van desde contenido y juegos en línea, hasta servicios de transacciones entre empresas o la automatización de sus fuerzas de ventas), haciéndolas disponibles en entornos de redes inalámbricas.

Existen muchos factores y comunidades que a mediano y largo plazo, determinarán el desarrollo y el mejor modelo de negocios a seguir en este segmento: en primer lugar, los proveedores de servicios y fabricantes de dispositivos y su relación con las comunidades de desarrolladores; en segundo lugar, los recursos de las redes inalámbricas: el ancho de banda usado para transmitir el volumen de información requerido (y obviamente su costo) en relación a la base de usuarios de la red. El potencial de las aplicaciones recae en las características de seguridad, flexibilidad, robustez, y costo de implementación de los estándares en que están basados (como por ejemplo IP, XML, Java, infraestructura de redes, estándares de autenticación, compresión de datos, etc.).

El mercado para los proveedores puede estar basado en varios modelos: puede únicamente proveer el servicio de acceso a una red; también puede proveer servicios de terceros; puede proveer servicios que la misma entidad posea o inclusive tomar el papel de desarrollador de servicios (mismos que puede ofrecer, o vender a otros proveedores). Dentro de todas esas modalidades, un proveedor puede o no ofrecer garantía sobre dichas aplicaciones.

Para las empresas se volverá atractiva la idea de automatizar las fases de los procesos de producción, distribución y comercialización de sus productos y servicios, haciéndolos a la vez más eficientes y más bajos en costos. Progresivamente, las empresas se verán involucradas en un ambiente de competencia más abierta y más equitativa; la disponibilidad, rapidez, accesibilidad y confiabilidad de la información será crítica para su permanencia en el mercado. Para los desarrolladores y fabricantes de dispositivos, será importante la demanda que tengan los servicios de red inalámbricos, de forma que se verán obligados a ofrecer aplicaciones y dispositivos con mayores ventajas, mejores prestaciones y un alto desempeño. En la misma medida que la demanda de aplicaciones y dispositivos crezca, irá creciendo la demanda y el valor de los servicios de los desarrolladores y los fabricantes.

Dentro de esta industria en pleno crecimiento y desarrollo, los usuarios finales de los servicios también saldrán beneficiados: tendrán comunicaciones, información, entretenimiento, aplicaciones y servicios en donde quiera que se encuentren, justo en el momento en que lo necesiten, y en cualquiera que sea el dispositivo del que dispongan. El mercado potencial, se extiende desde el hogar hasta las grandes empresas, y donde sea que la disponibilidad de ésta información (que se puede originar en prácticamente cualquier lugar del mundo, en la oficina, el banco, en otra ciudad, en provincia o en el extranjero) signifique un posible beneficio.

Conclusiones

Existen hoy día varias tecnologías para la comunicación inalámbrica y la tendencia indica que esto crecerá vertiginosamente en el futuro a corto plazo. Se están preparando ya nuevas tecnologías que se adoptarán como estándares que mejorarán la velocidad y alcance de las redes inalámbricas, con lo que no hará falta más que encender nuestros equipos de cómputo dentro de nuestros hogares para estar conectados a una red de cobertura global. De las opciones de comunicación inalámbrica para teléfonos celulares, la más prometedor y brillante es 3G, que se apunta como la tecnología que cambiará el modo en que empleamos los teléfonos celulares y lo que hacemos con ellos para comunicarnos.

La nueva generación de equipos telefónicos brindará soporte para Java, con lo que las aplicaciones creadas tendrán un mayor valor e importancia. En el campo de las LAN's inalámbricas hay dos tecnologías que se han vuelto muy populares.

Bluetooth, si bien no parece ser una tecnología indicada para montar redes de cómputo, sí lo es para implementar una solución apropiada para enlazar algún dispositivo a una red ya existente, con su bajo consumo de energía y reducido tamaño se puede integrar a dispositivos como teléfonos celulares y PDA's.

Si lo que se quiere es montar una red local a alta velocidad la opción más indicada es el estándar 802.11g, ya que tiene mayor velocidad que su antecesor 802.11a pero con un alcance mayor, además de ser compatible con varios estándares 802.11 anteriores. Cualquiera de las tecnologías mencionadas permite la comunicación en red de aplicaciones J2ME.

Se espera el surgimiento y consolidación de nuevos estándares de comunicación inalámbrica (específicamente WUSB y 802.11g). Seguramente cada estándar ocupará un lugar, un mercado y determinada aplicación debido a sus propias características; unos estándares se utilizarán más que otros, puede ser que algunos desaparezcan, sean mejorados y optimizados o simplemente abandonados. Lo que sí es seguro, es que la conectividad de redes inalámbricas irá creciendo y seguirá desarrollándose. No es difícil pensar que en un futuro a mediano plazo ya viviremos en un mundo sin cables.

Gracias a los avances tecnológicos que nos permiten disponer de una red de datos en todo momento, y las tecnologías estándares que facilitan la interconexión de sistemas en un entorno heterogeneo, la disponibilidad de cualquier tipo de información en cualquier lugar y el cualquier momento se convertirá en un escenario cotidiano.

En el siguiente capítulo, se presentará el desarrollo de un sistema programado en lenguaje Java que utiliza Servicios Web para la comunicación con dispositivos portátiles a través de redes inalámbricas. Estas tecnologías hacen que el sistema sea inherentemente flexible, modular, extensible, escalable, abierto, basado en estándares e independiente de cualquier plataforma.

4.

Desarrollo

4.1. Antecedentes

Hoy en día, ya no resulta extraño escuchar en los diversos medios de comunicación acerca de los delitos que se comenten con las tarjetas de crédito y débito en los restaurantes, así como los reportes que indican las pérdidas económicas provocadas a estos servicios por sus propios empleados a través de fraudes en caja.

Los delitos más comunes en los restaurantes son el cargo indebido y la clonación de tarjetas, debido a que los clientes generalmente pierden de vista por un momento su dinero plástico. Esto da oportunidad a que la tarjeta sea clonada o cargada de manera indebida, siendo el gran afectado el cliente. Otro delito que afecta a los restaurantes, es el robo por parte de los empleados que tienen contacto directo con el efectivo, quienes simplemente aprovechan cualquier oportunidad para retirar dinero que no registran en el sistema. Por ello, no hay manera de saber de forma precisa, a cuánto ascienden las pérdidas económicas ocasionadas en este caso al restaurante.

La clonación de tarjetas es la actividad más peligrosa y lucrativa, siendo un delito que ha crecido en los últimos años gracias a que los equipos empleados para realizarlo se han hecho más pequeños y más baratos. Conocido como "skimmer", este dispositivo es tan pequeño que puede disfrazarse como un localizador o una funda de teléfono y de esta forma portarse en el cinturón, con lo que sólo bastan unos cuantos segundos para clonar una tarjeta. Debe aclararse que la clonación de tarjetas no es la actividad de obtener un duplicado idéntico en plástico del original, sino al obtener y copiar el contenido de la banda magnética donde reside toda la información de la cuenta.

Según cifras de la CONDUSEF (Comisión Nacional para la Protección y Defensa de Usuarios) las cifras de delitos cometidos en el inicio del 2004 han aumentado respecto a 2003 y este a su vez aumentó con respecto a 2002, mostrando ya cifras alarmantes de este fraude que ocasiona pérdidas millonarias a los tarjeta habientes y aunque no afecta directamente a los dueños de los restaurantes, si les afecta en cuanto a imagen y prestigio, ya que los clientes que han sido defraudados por clonación de tarjetas tendrán presente en su mente que fue en cierto lugar y optarán por acudir a otro restaurante, causando la pérdida de clientes del negocio.

Desarrollo

A continuación se muestra un reporte presentado por CONDUSEF sobre las reclamaciones presentadas en 2002 y se le anexan datos preliminares de 2003 y 2004.

Enero - Diciembre 2002			
Institución Financiera	No. Tarjetas ^A	Reclamaciones en CONDUSEF	Perdidas económicas en pesos[†]
Serfin	121,645	389	---
Afirme	5,209	3	---
Bital	310,508	225	---
Santander Mexicano	119,548	68	---
Mercantil del Norte	234,150	80	---
Ixe	2,741	1	---
BBVA Bancomer	3,019,511	1,265	---
Banamex	4,384,863	1,363	---
Scotiabank Inverlat	214,669	60	---
American Express	432,492	38	---
Inbursa	290	0	---
Total	8,845,626	3,492	627,000,000

Enero - Diciembre 2003			
Institución Financiera	No. Tarjetas ^A	Reclamaciones en CONDUSEF	Perdidas económicas en pesos
Total	12,408,000	4,000+	715,000,000

Primer Bimestre 2004			
Institución Financiera	No. Tarjetas ^A	Reclamaciones en CONDUSEF	Perdidas económicas en pesos
Total	1,306,000	5% incremento respecto a primer bimestre 2003	30,000,000 incremento respecto a primer bimestre 2003

^A Datos obtenidos del Sistema de Análisis Financiero (SAF) de la CNBV, Abril de 2004.

Cabe señalar que el número de reclamaciones presentadas por CONDUSEF corresponde a un estimado del 10% del total de fraudes por copiado de tarjetas bancarias. No todos los delitos son denunciados y en muchas ocasiones los mismos tarjeta habientes no se percatan de cobros indebidos en sus estados de cuenta. Esto hace muy difícil determinar cual es el total exacto de delitos por clonación de tarjetas, de forma que los números reales son aún más alarmantes.

[†] Los datos de pérdidas económicas por institución son confidenciales y no son proporcionados por CONDUSEF.

Debido a la gran cantidad de delitos que la CONDUSEF y los bancos deben analizar y teniendo en mente que no todas las reclamaciones proceden, los bancos han determinado cobrar comisiones por cada reclamación improcedente, lo que agrega una molestia más para los clientes y una desconfianza mayor hacia los restaurantes que no muestran de forma clara el cobro con tarjetas, así que la falta de memoria puede ocasionar aún más pérdida de dinero. La comisión más cara por este concepto es de American Express, en el que si la reclamación por un consumo menor a los 6 mil pesos resulta improcedente (es decir, que la firma de un voucher sí corresponde a la firma del tarjeta habiente), el cargo será de \$30.00USD más IVA. A continuación se muestra una tabla con los costos de comisión por reclamación improcedente de algunos bancos.

Costos de comisión por reclamación improcedente	
Institución Financiera	Costo en pesos (mas IVA)
American Express	330.00
Bital	320.00
Ixe	250.00
BBVA Bancomer	200.00
Scotia Bank	180.00
Banorte	174.00
Santander Serfin	170.00
Banco del Bajío	100.00
Banamex	50.00

Como se mencionó, el otro delito más común en los restaurantes es el que se lleva a cabo por los mismos empleados (que pueden ser meseros, cajeros e incluso cocineros). El delito se lleva a cabo tomando dinero de la caja sin registrar la entrada del efectivo, de forma que los empresarios de restaurantes deben gastar dinero extra para controlar de manera más eficiente la contabilidad y seguridad financiera del negocio. Sin embargo, este delito es difícil de seguir y de eliminar ya que los mismos empleados se encubren y las sumas robadas son pequeñas pero constantes, sin tomar en cuenta que los dueños de restaurantes no pueden mostrar tal desconfianza de sus empleados.

Adicionalmente existen situaciones que afectan la rentabilidad de un negocio por el poco control o mal manejo que se hace de aspectos como la imagen del establecimiento, la satisfacción del cliente, la calidad del servicio, la disponibilidad de insumos, etc. Este control interno, que comúnmente se efectúa de manera manual, reduce la productividad y aumenta el margen de error en los procesos. Por ejemplo: regularmente los restaurantes llevan a cabo su contabilidad a través de las notas de consumo, lo que lo hace una actividad tediosa, repetitiva y propensa a errores. Otro problema ocurre en la cocina, donde en ocasiones no se lleva un estricto orden de lo que más se consume y cuales son las necesidades más inmediatas; si se llegan a llevar a cabo estas actividades, resulta ser más por experiencia e intuición que por una metodología bien definida.

4.2. Solución propuesta

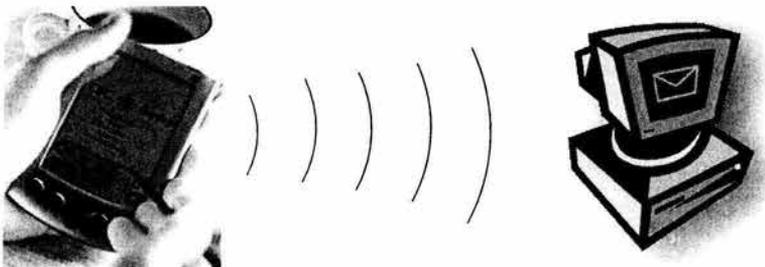
Sin duda el punto más crítico y que requiere atención inmediata es el de evitar los delitos por clonación de tarjetas, que afecta tanto a clientes como al establecimiento, ocasionando pérdidas económicas y de confianza en el negocio.

Para resolver este problema se ha propuesto desarrollar un sistema de cómputo que evite en la medida de lo posible el control manual de los procesos involucrados, que muestre en todo momento de forma clara y sencilla la información relacionada, además de llevarse un registro completo de todo lo que sucede dentro del establecimiento.

A través de este sistema se mejoraría la seguridad y se agilizarían los procesos, logrando así una mejor productividad y mayores ganancias. Se llevaría un registro de todas las actividades realizadas que nos permitirían un análisis del negocio e incluso un control de la calidad. El sistema sería fácil de implementar y de usar, permitiendo adecuarlo a los procesos específicos de cada negocio, además de permitir crecer y expandir su funcionalidad debido a su modularidad, así como poder ser usado por negocios relativamente pequeños o por grandes cadenas de restaurantes.

Para evitar el robo por clonación, se eliminaría el contacto físico de los meseros y personal del restaurante con las tarjetas bancarias, permitiendo que el cliente nunca pierda de vista, e incluso ni la posesión de sus plásticos. El mesero entonces, únicamente se dedicaría a recibir las órdenes de los clientes por medio de un dispositivo de cómputo pequeño y portátil que fuera capaz de desplegar y seleccionar los platillos de un menú, enviando la información a un servidor central donde se registrará una bitácora de información y procesará adecuadamente para atender la orden del cliente.

Dicho dispositivo de cómputo puede ser encontrado en los PDA que en los últimos años han cobrado mucha popularidad y sus costos se han reducido de manera muy importante, además, su poder de cómputo es más que suficiente para realizar dichas tareas, sin mencionar sus grandes capacidades de comunicación. Existen muchas marcas de PDA's, pero la gran mayoría están basadas en sólo dos Sistemas Operativos: PALM y Windows Mobile.



La forma de comunicación sería inalámbrica y para ello ya existen varias tecnologías disponibles en el mercado y de un costo razonablemente bajo. Entre estas tecnologías podemos encontrar:

- Bluetooth
- Wi-Fi (802.11)
- IrDA

Adicionalmente, en un futuro contaremos con WUSB, tecnología que está en vías de desarrollo.

La información generada por el dispositivo portátil inalámbrico sería recibida por un equipo de cómputo central funcionando como servidor, que es dónde residiría la aplicación maestra y donde se haría todo el procesamiento de información para su posterior aprovechamiento. Constaría de un sistema de cómputo con capacidad de emitir y recibir información a través de un protocolo inalámbrico seguro que tendría una Base de Datos para almacenar y extraer su información. La comunicación entre el dispositivo portátil inalámbrico y el servidor sería en tiempo real, lo que agilizaría el proceso de atención al cliente, reduciría los tiempos de preparación de los alimentos y mejoraría el proceso de cobro al realizar la cuenta de manera automática.

Otro aspecto que se considera es la generación de información que permita hacer un análisis del negocio, es decir, llevar estadísticas de comportamiento y de recursos, lo que permitiría posteriormente ver oportunidades de mejora. La contabilidad sería beneficiada debido a que el sistema llevaría los registros de cada una de las órdenes y pedidos atendidos, de forma que podría presentar informes diarios, semanales y anuales de la actividad económica del negocio, todo ello sin requerir de infraestructura tecnológica adicional.

4.3. Implementación

4.3.1. Sistema SIR

Llamado SIR por sus iniciales (Sistema de Información para Restaurantes), el sistema está desarrollado con tecnología Java. Se decidió emplear este lenguaje debido a que brinda muchas bondades de desarrollo e implementación, pero la razón más importante han sido las posibilidades de la variante de Java para dispositivos portátiles, J2ME (Java 2 Micro Edition) que en años recientes ha tenido gran aceptación en la industria y cada día más dispositivos electrónicos brindan soporte para Java: desde sistemas inmersos hasta teléfonos celulares y organizadores personales digitales. Las ventajas de emplear Java también radican en la posibilidad de emplear diferentes plataformas de hardware y software para su ejecución, dando así, la oportunidad de elegir la plataforma que más agrade o convenga al negocio. Otra tecnología que se decidió emplear para el desarrollo del sistema y que es de vital importancia son los Web Services, una tecnología de última generación y de reciente creación que se apunta para ser la plataforma de desarrollo del futuro cercano. Como ya se explicó en capítulos anteriores, los Web Services son aplicaciones que residen en la Red y que permiten la comunicación entre sistemas independientemente de su arquitectura, comunicándose a través del protocolo http. Al emplear Web Services, se crean las posibilidades de crecimiento del sistema, además de presentar la oportunidad de iniciar un nuevo modelo de negocio en el que el dueño de la aplicación puede proveer el servicio a diferentes restaurantes.

El sistema está desarrollado bajo un modelo de 3 capas bien establecidas con la finalidad de hacerlo modular y mejorar así el mantenimiento y desarrollo de nuevas extensiones. La capa más importante es la capa de negocios que es expuesta al exterior como un Web Service y es la base del sistema. A continuación se detallan las 3 capas que conforman el sistema y que construyen un sistema de Web Services accesado por clientes inalámbricos.

Sistema de Información para Restaurantes

Capa del Sistema	Descripción
Capa Persistente	Capa donde reside la Base de Datos o algún otro medio de almacenamiento. Los accesos son realizados a través de EJB's de J2EE
Capa de Negocios	Desarrollado con tecnología J2EE, incluye todas las reglas y procesos aplicados a la información recibida. Incluye control de transacciones y seguridad del sistema. Expuestos al exterior como WebService.
Capa Cliente	Clientes capaces de comunicarse con el Web Service y obtener la información necesaria; además se pueden conectar a una red alámbrica ó inalámbrica. Desarrollados con tecnología J2SE y J2ME.

Capa Persistente

Tiene como finalidad dar persistencia a los datos que ingresa y procesa el sistema, almacenando de forma permanente toda la información relevante para el control del negocio y llevar a cabo un análisis posterior que permita optimizar los recursos y mejorar la calidad de los servicios.

Existen varias formas de almacenar y hacer persistente la información, pero la más ampliamente aceptada es una Base de Datos (*RDBMS*), accesada a través de una aplicación desarrollada en algún lenguaje de programación. En este caso la aplicación encargada de la persistencia está desarrollada con tecnología Java 2 Enterprise Edition, empleando sus EJB's como medios de ingreso y consulta de datos. Al emplear EJB's (Entity Beans) para la aplicación, estamos aprovechando la bondad que brinda J2EE de hacer independiente la capa persistente del RDBMS, con lo que podemos elegir cualquier manejador de base de datos dependiendo de nuestras necesidades, ya sea una base de datos comercial y de gran aceptación como lo es Oracle o una base de datos de Software Libre como lo son PostgreSQL o MySQL, la elección dependerá de las necesidades de almacenamiento y del presupuesto disponible.

Debido a que se empleará J2EE para el desarrollo de la capa persistente, es necesario el empleo de un contenedor para la aplicación, lo cual podemos encontrar en los Servidores de Aplicaciones, que son sistemas de cómputo complejos encargados de llevar a cabo el control de las transacciones, la seguridad y estabilidad de una aplicación, liberando al desarrollador de programar dichas características explícitamente. Haciendo uso de un Servidor de Aplicaciones se reducen los tiempos de desarrollo y se incrementa la portabilidad de un sistema de cómputo, siendo las limitaciones del Servidor de Aplicaciones las que pueden llegar a comprometer la portabilidad de nuestra aplicación. Existen muchos Servidores de Aplicación entre los cuales podemos mencionar

- BEA WebLogic
- IBM Websphere
- Oracle 9iAS
- JBoss Application Server
- Borland Enterprise Server
- Novel exteNd
- SUN Application Server

Capa de Negocios

En la capa de negocios (business tier) es donde se lleva a cabo el procesamiento de la información de entrada para dar un resultado y mostrarlo al usuario a través de una capa de presentación. Pensemos en la capa de negocios como el circuito de procesamiento de una calculadora que se encarga de realizar operaciones matemáticas a datos proporcionados por un teclado y que ya una vez obtenido el resultado lo entrega a una pantalla para que sea mostrado al usuario.

Desarrollo

En el caso de nuestra aplicación para Restaurantes, la capa de negocios recibiría los datos de una orden y se encargaría de asegurarse que esos datos fueran entregados al personal de cocina, llevar a cabo un seguimiento del estatus de dicha orden y registrar cuando ya esté preparada y lista para ser servida a los comensales, también se encargaría de llevar a cabo la contabilidad del negocio y de llevar una bitácora del comportamiento de las operaciones realizadas en el transcurso de un día, una semana, un mes, etc. En la capa de negocios estaría ubicado el núcleo, que proporciona la funcionalidad más importante y básica al sistema, la parte medular de la aplicación y por tanto el módulo más importante y crítico, por lo que resulta de vital importancia que la confiabilidad y exactitud sean cercanas al 100%.

Al igual que la capa persistente, la capa de negocios aprovecha la tecnología J2EE, y por medio de EJB's (session beans) se encarga de llevar a cabo las operaciones relacionadas con los procesos del negocio. Además, toma provecho de las bondades del Servidor de Aplicaciones y por medio del contenedor se realiza el balanceo de carga y el control de la seguridad del sistema. La única diferencia con otros sistemas radica en que se hace empleo de la tecnología de los Web Services para mostrar la aplicación al exterior, de forma que pueda ser aprovechada por cualquier cliente Web Service autorizado, ya sea accediendo a la red por medios inalámbricos de manera local o accediendo remotamente desde una red convencional conectada al Internet. El empleo de Web Services nos da otra ventaja, y resulta muy importante a la hora de extender las capacidades del sistema; debido a que la información que recibe la capa de negocios debe ser a través de un protocolo estándar establecido por los Web Services, en este caso el protocolo es SOAP, cualquier cliente programado en cualquier lenguaje y arquitectura (mientras siga este protocolo estándar) podrá tener comunicación con el Web Service de la capa de negocios. SOAP es un protocolo de comunicación basado en XML por medio del cual se pueden hacer peticiones y recibir resultados de servicio a un Web Service.

Capa Cliente

La capa cliente es una interfaz por medio de la cuál se ingresa información que es enviada a la capa de negocios y después de que ésta haya realizado las operaciones correspondientes recibe un resultado para ser desplegada al usuario, en este caso el mesero, cajero o personal de cocina. El cliente debe cumplir con ciertos requerimientos de implementación como lo es el protocolo de comunicación empleado para hacer llegar la información al Web Service y recibirla. El protocolo que debe implementar es SOAP, con el que debe hacer llegar objetos que contengan la información generada, en este caso los objetos transportados son de tipo String, es decir, cadenas de caracteres.

Se decidió emplear únicamente objetos de tipo String debido a que los dispositivos que se emplearán como clientes (asistentes personales digitales) tienen aún limitaciones de memoria y velocidad de transmisión y procesamiento de datos, lo que nos limita respecto a los tipos de objetos que puede emplear.

Existen 3 tipos de clientes que serán utilizados en el Sistema de Información para Restaurantes, y cada uno de ellos está orientado a satisfacer una necesidad específica, los clientes y su uso son los siguientes:

Sistema de Información para Restaurantes

Tipo de Cliente	Descripción
Cliente de Mesero	El cliente estará dentro de un dispositivo portátil inalámbrico (Palm) que tenga capacidad de conectarse de manera inalámbrica a la red. Por medio de su interfaz podrá generar órdenes a través de un menú desplegado y los enviará al Web Service. También puede ser una página Web lo que permitiría ordenar en línea por Internet. Está programado en J2ME.
Cliente de Cocina	Estará ubicado en un equipo de cómputo provisto de "touchscreen" (preferiblemente), y por medio de una conexión a red tradicional solicitará información al WebService, relacionada con las órdenes que se levantan y detalles de los platillos que se deberán preparar, así mismo podrá notificar el estatus de preparación. Está programado en J2SE.
Cliente de Caja	Aplicación cliente que estará en un equipo de cómputo de bajo costo y que se encargará de hacer los cobros de las cuentas realizadas. Solicitará al Web Service el total de la cuenta y la información detallada para el cliente, realizará los cobros por el medio deseado: efectivo, vales o tarjeta. Está programado en J2SE.

Cliente Mesero (Restaurante)

El cliente mesero tiene la función de ser una interfaz de entrada de datos por medio de la cual se generan las órdenes de platillos y comidas que ha ordenado el cliente. El cliente, por tanto, debe ser un dispositivo que pueda llevar consigo el mesero y que sea de fácil uso, para ello se hace uso de un dispositivo Palm o WindowsCE que tenga capacidad de conectarse a la red de manera inalámbrica, ya sea por Bluetooth o Wi-Fi que son las dos tecnologías más usadas, no importa cual se use, debido a que el sistema es independiente de la capa de red gracias a Java. Para desarrollar la aplicación cliente se determinó emplear el lenguaje *Java 2 Micro Edition*, debido a que es posible correr aplicaciones creadas en este lenguaje en cualquier dispositivo siempre y cuando disponga de una máquina virtual de Java *KVM* que soporte el perfil *MIDP*.

Por medio de mensajes de SOAP el cliente de Mesero haría llegar la información al Web Service, dicha información estaría compuesta por el tipo de platillo que ha solicitado el comensal, la cantidad de estos y las bebidas deseadas. La comunicación es de manera instantánea, se puede decir que en tiempo real (salvo por la *latencia* de la red) lo que agilizaría el proceso de atención de la orden. En cuanto la orden esté lista, se puede emitir un mensaje al mesero para que recoja los platillos preparados en la cocina. Para llevar un control y saber a quien pertenece dicha orden se genera un "ticket" en el servidor para identificar la mesa, el mesero, los alimentos y la mesa asignada, dicho ticket será posteriormente empleado por el cliente de Caja para hacer el cobro correspondiente.

Cliente Mesero (Servicios Adicionales)

El uso de un modelo de tres capas para separar las reglas de negocios, la base de datos y los clientes, nos da una gran libertad respecto a los clientes que se pueden programar. Como se había mencionado antes, por medio de un nuevo cliente se puede implementar un sitio en Internet para hacer pedidos en línea de comida para llevar, lo único que se debe hacer es seguir el protocolo de SOAP y la definición del Web Service, este último está especificado en un documento WSDL.

El desarrollo de dicho sitio en Internet para ordenar en línea puede ser elaborado en cualquier lenguaje de programación lo que da la libertad de elegir el que más seguridad brinde o el que sea más rápido para desarrollar, de forma que podríamos generar un sitio Web hecho en ASP, JSP, PHP, etc.

En el sitio Web se muestran los platillos existentes que serán extraídos de la BD y el usuario tiene la libertad de elegir que y cuantos va a ordenar, esta información al igual que se hace el cliente Palm, será enviada al Web Service (capa de negocios) a través de un mensaje de SOAP que contiene objetos tipo String. En seguida, la página Web mostrará el total de la cuenta y realizará el cobro con tarjeta de crédito. Previamente el usuario ya habrá ingresado la información para hacer la entrega de los platillos en su domicilio, inmediatamente después de ordenar serán preparados como si fueran platillos para una mesa del restaurante con la única diferencia de que tendrán una descripción de "para llevar" de forma que al cambiar el estatus de un platillo a "listo" en lugar de llevar a la mesa se entregará empaquetado al repartidor.

Cliente Cocina

Tiene la función de ser una aplicación de visualización para la cocina. Por medio de ésta aplicación se indicará al jefe de cocina los alimentos que se están ordenando y en orden de petición serán mostrados en una pantalla donde además podrá ir indicando el estatus de cada uno de los platillos (*abierto, en preparación y listo*) de forma que el mesero tenga siempre la información de cómo va la orden y se le avise cuando ya puede ir por los alimentos.

El cliente de cocina mantiene una comunicación constante con el Web Service y por medio de mensajes de SOAP va solicitando a la capa de negocios información sobre las órdenes abiertas y le envía respuestas para cambiar los estatus.

En cuanto se abre una orden aparece en pantalla la notificación con la información de dicha orden, como los platillos, la cantidad y alguna nota en especial. Para cuando el encargado de interactuar con dicha aplicación cliente indica que ya está listo, la orden es retirada de pantalla.

Se recomienda que para la aplicación de cliente de cocina se disponga de un monitor de tipo "touch screen" de forma que el ingresar los estatus de los platillos sólo sea cuestión de tocar un punto en la pantalla, caso contrario se deberá hacer uso de un ratón lo que no será muy práctico ni higiénico ya que se tiene contacto con los alimentos.

Cliente Caja

El cliente de caja es una interfaz por medio de la cuál se puede consultar el total calculado para una orden y permite hacer el cobro de la misma ya sea a través de efectivo, vales de restaurante o por pago con tarjeta. Para tener acceso al cliente de caja se requiere de un nombre de usuario y contraseña, bajo la restricción de que sólo las personas responsables de la caja tendrán acceso.

Cuando algún cliente solicita la cuenta, el mesero le proveerá de su número de "ticket" y le pedirá que pase a la caja para realizar el pago, con esto el mesero no participará en el cobro, se dedicará únicamente a servir alimentos lo que motiva que el cliente jamás pierda posesión de su tarjeta. Al llegar al área de caja el responsable introducirá el "ticket" en la aplicación e inmediatamente se mostrará en pantalla la información relacionada con la orden, un detallado de alimentos, la cuenta total y parcial, así como un módulo para hacer el cobro por efectivo y tarjeta. Es importante señalar que como una medida de seguridad, el cobro por tarjeta se hará a la vista del cliente quien incluso puede ser el que la deslice la por el sistema de lectura proporcionado por el banco para hacer el cobro.

La información es regresada al Web Service, donde la capa de negocios procesará la información, registrará en la BD los movimientos que se realizan y devolverá el resultado de la operación al cliente. Esta información se puede utilizar para un posterior análisis del negocio.

4.3.2. Escenario

A continuación se desarrollará una descripción de la funcionalidad del sistema en un ambiente real, indicando a cada momento los procesos involucrados y su interacción con el sistema SIR. Previamente se debió configurar la información del menú en la base de datos, así como los usuarios y sus respectivos roles.

El proceso inicia al encender la aplicación de cliente de mesero que lo primero que mostrará será una pantalla de inicio de sesión donde el usuario debe introducir su nombre de usuario y contraseña.[‡]



Al haberse registrado correctamente el usuario, el dispositivo solicitará al Web Service que le proporcione la información relacionada con el menú que el restaurante tiene para ese día y será almacenado en la memoria persistente del dispositivo Palm.



[‡] En este punto, cabe aclarar que algunos aspectos y características de la GUI del cliente del mesero fueron determinados por ciertas limitaciones de la especificación 1.0 de MIDP. A la fecha de liberación del Sistema, la implementación de MIDP 2.0 de IBM (WebSphere Micro Environment) aún se encontraba en fase de desarrollo.

Cuando un cliente es acomodado en alguna mesa, el mesero atenderá y solicitará que le indiquen lo que desean ordenar.

Cuando el mesero abra una orden en una mesa, creará en el servidor un número de "ticket", que será el identificador empleado dentro del sistema. A través de la pantalla del dispositivo Palm, el mesero podrá ir seleccionando los platillos que se le solicitan (ayudado por el contenido del menú que previamente se descargó), con sólo dar clicks en los ítems mostrados.

Auto-gener...Nuevo Elemento

Cant.: 1

Descr.: Carne Asada

010:Carne Asada

Nota: Bien cocida

Atras OK

En el instante que se acepta la orden, el cliente de mesero envía la información al Web Service, que registrará esta acción en la base de datos, identificándolos a través del "ticket" generado. En la aplicación cliente de mesero, se tendrá una sesión abierta donde se podrán agregar más alimentos, y para cuando los comensales decidan pedir la cuenta, el mesero sólo tendrá que ejecutar la acción de "cerrar orden".

Auto-generated-Ver Mesas

↔ Mesa 01 Libre

↔ Mesa 03 Libre

↔ Mesa 04 Libre

↔ **Mesa 05 * Sinc.**

↔ Mesa 07 Libre

↔ Mesa 11 Libre

↔ Mesa 19 Libre

Logout Sinc. Cerr.

Desarrollo

Por otro lado, al momento de generar la orden de algún platillo, la información que se registró en la base de datos por parte del Web Service es solicitada por la aplicación cliente de cocina tras haber recibido la correspondiente notificación. En base a esa información, mostrará en pantalla los platillos que se deben preparar. Cualquier orden nueva mostrada en pantalla tendrá el estatus de "ESPERANDO" lo que indica que no ha sido aún atendida por el jefe de cocina.

SIR V.1.0. - K System Client (Cocina)

Archivo Ayuda

CONECTADO AL SERVIDOR

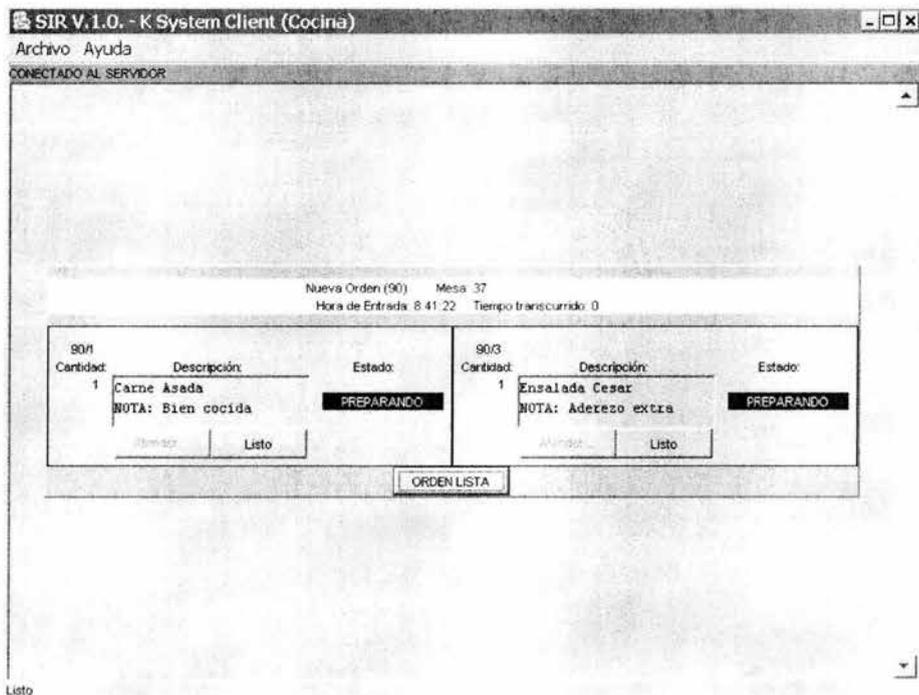
Nueva Orden (90) Mesa: 37
Hora de Entrada: 8:41:22 Tiempo transcurrido: 0

90/1	Descripción:	Estado:	90/3	Descripción:	Estado:
Cantidad: 1	Carne Asada	ESPERANDO	Cantidad: 1	Ensalada Cesar	ESPERANDO
NOTA: Bien cocida			NOTA: Aderezo extra		
Atender... [] []			Atender... [] []		

ORDEN LISTA

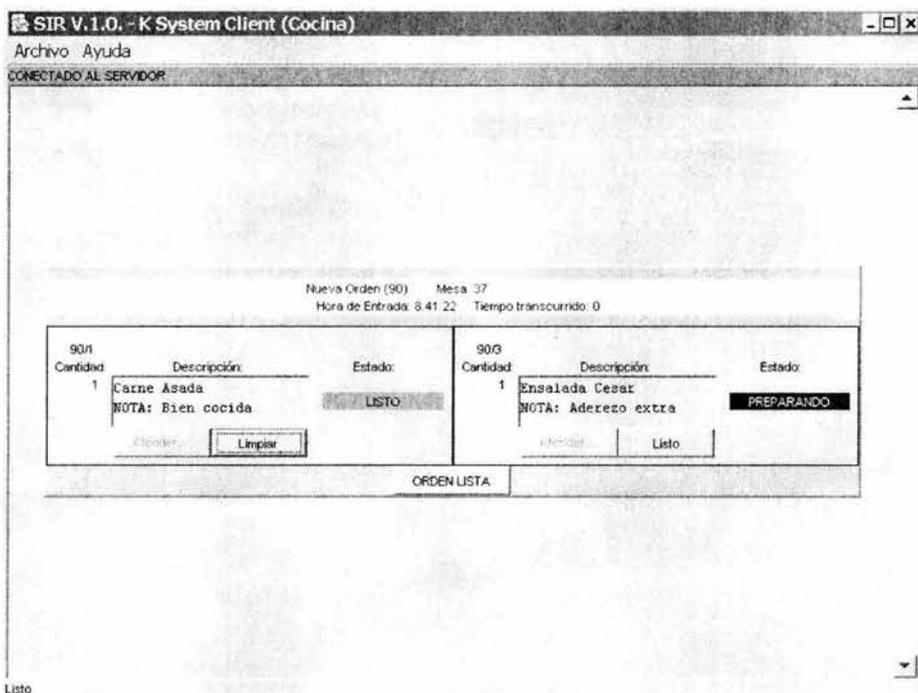
Listo

Cuando el jefe de cocina es notificado a través del cliente de cocina, de que hay una nueva orden esperando (con platillos denotados con el estado de "ESPERANDO"), indica que se prepare y le da un clic al botón de "Atender", con lo que se notificará al servidor y el estado del platillo cambiará a "PREPARANDO".



Desarrollo

Cuando el platillo está listo el jefe de cocina acude de nueva cuenta a la pantalla del monitor de la aplicación cliente de cocina para dar clic de nueva cuenta y determinar el estatus del platillo como "Listo", con lo que se le avisa al Web Service que se han terminado de preparar los alimentos, esto se registra en la BD y se emite el aviso al mesero para que pase a recoger el o los platillos. Este aviso se dirigía originalmente al dispositivo portátil, que vibraba y emitía un sonido para notificar al mesero, sin embargo, el costo en degradación de desempeño del cliente al tener un *demonio* escuchando notificaciones en un puerto era muy alto. Además no se podía garantizar que el mesero se diera cuenta de la notificación. El esquema propuesto, es una pantalla a la vista de todos los meseros que despliegue uno tras otro el número de mesa de la orden que está lista y/o la descripción de los platillos que se vayan preparando.



La información referente a dicho ticket, se desplegará en pantalla, incluyendo los alimentos que se ordenaron y el total de la cuenta.

SIR V.1.0. - C System Client (Cocina) [_] [□] [X]

Archivo Ayuda

CONECTADO AL SERVIDOR (Usuario: gerardo)

Información de Cuenta Mesa: 20 Mesero: 1 (Saul Castro Overa)

ID: Hora de Entrada: 08:33:29 Tiempo de Estancia: 09:33:30

Resultado: Datos de Orden Encontrada (0000000095)

Cantidad	Descripción	Precio	Cargo
1	001 - Frij. Ch. (ord)	12.00	12.00
1	003 - Ens. C.	17.00	17.00
1	008 - Club snchw	22.50	22.50
Total		51.50	

Forma de Pago:

Efectivo Monto Total a Pagar: 51.50

Tarjeta

Vaso Cantidad Recibida: \$

Moneda Dura Cambio: \$ -51.50

American Express Requiere Factura RFC: Dirección:

Vale de Restaurante

Usuario: gerardo (ID: 4)

Hora de Inicio de Sesión: 05:09 PM

Listo

Desarrollo

El cliente pagará, ya sea con efectivo, con vales o con tarjeta y para ello se emplea la aplicación donde se incluye una pequeña calculadora que con solo introducir la cantidad recibida indicará el cambio a entregar.

The screenshot shows the 'SIR V.1.0. - C System Client (Cocina)' application window. At the top, it displays 'Archivo Ayuda' and 'CONECTADO AL SERVIDOR (Usuario: gerardo)'. Below this, there is a section for 'Información de Cuenta' with an 'ID:' field and a 'Cancelar' button. To the right, it shows 'Mesa: 20', 'Mesero: 1 (Saul Castro Olvera)', 'Hora de Entrada: 06:33:29', and 'Tiempo de Estancia: 09:39:30'. A status bar indicates 'Resultado: Datos de Orden Encontrada (0000000095)'. The main area contains a table with the following data:

Cantidad	Descripción	Precio	Cargo
1	001 - Frij. Ch. (ord)	12.00	12.00
1	003 - Ens. C.	17.00	17.00
1	006 - Club sndhw	22.50	22.50
	Total	51.50	

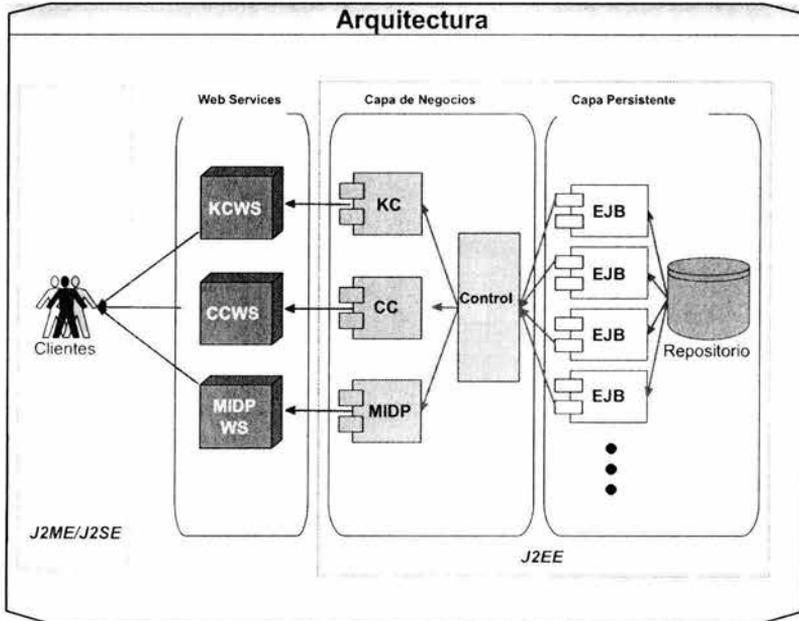
Below the table, the 'Forma de Pago:' section offers several options: Efectivo, Tarjeta, Vain, Moneda Camb, Cheque en Efectivo, and Vole de Restaurante. The 'Monto Total a Pagar: \$1.50' is displayed. For the 'Tarjeta' option, there is a 'Número de Tarjeta de Crédito:' field, a 'Cantidad Recibida: \$ 100.' field, and a 'Cambio: \$ 48.50' calculation. A 'Requerir Factura' checkbox is checked. The 'RFC' is 'SEI 950623-RT7' and the 'Dirección' is 'Seguridad Integral S.A., Calzada del Sol #28, Colonia El Arenal, C.P. 07654'. At the bottom, there are 'Pagar' and 'Cancelar' buttons, and a status bar with 'Usuario: gerardo (ID: 4)', 'Hora de Inicio de Sesión: 05:09 PM', 'Bloquear Sesión', and 'Cerrar Sesión' buttons. The word 'Listo' is visible in the bottom left corner.

Después de haberse realizado el cobro de manera exitosa, se imprimirá una nota de consumo que se entregará al cliente donde se muestra el detallado de los alimentos ordenados. Finalmente, se envía esta confirmación al Web Service donde se cambiará el estatus de la orden a pagado.

Cabe señalar que para realizar el cobro por tarjeta, el mismo cliente puede ser quien la deslice por el dispositivo para cobro, de forma que en todo momento el cliente tuvo en sus manos y a la vista su dinero plástico.

4.3.3. Arquitectura

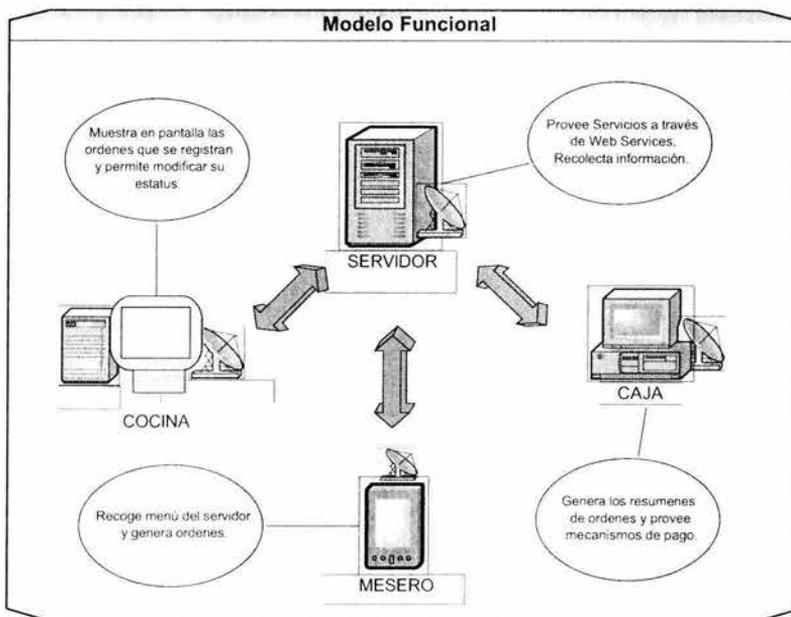
En el sistema SIR, se contempló la siguiente arquitectura, con el objetivo de hacerla lo más flexible, modular y extensible que fuera posible, haciendo la lógica de programación de cada capa independiente de las demás capas.



Se tiene el acceso a Base de Datos independientemente del manejador que se utilice por medio de Entity Beans. En esta implementación se utilizó un controlador JDBC para PostgreSQL DataBase Server 7.4.1. La lógica del acceso a datos se hizo mediante un Session Bean, utilizado a su vez por 3 Beans más, que implementan la lógica del negocio. Estos Beans son publicados a través de 3 servicios Web (1 por cada cliente) a los usuarios.

Desarrollo

El modelo del sistema una vez implementado, se puede ver de la siguiente forma:



El cliente del mesero está encargado de enviar los mensajes al servidor para generar y llevar el control de las órdenes, mismas que son constantemente actualizadas en el cliente de la cocina. Una vez que la orden se cierra, su pago se efectuará a través del cliente de caja.

La siguiente figura corresponde al diagrama de clases de la arquitectura del servidor, desde los Entity Beans, a los Web Services.

Desarrollo

4.3.4. Ventajas y Costos

Ventajas

El Sistema de Información para Restaurantes permite facilitar la operación y administración del negocio, convirtiéndose en una ventaja competitiva respecto a negocios que se administran de la manera tradicional.

El SIR permite un mejor control de los procesos con el fin de mejorar la calidad y operación.

Administrativamente se pueden implementar mecanismos de automatización de tareas diarias como contabilidad y administración del inventario, lo que significa ahorro de costos y optimización de la inversión.

Otras ventajas, serían:

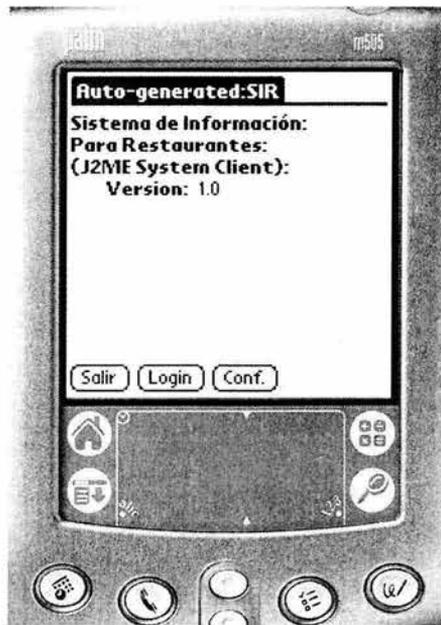
Sistema de Información para Restaurantes ventajas operativas y administrativas	
Ventaja	Descripción
Mejor atención al cliente	<ul style="list-style-type: none">• Con el empleo del dispositivo inalámbrico las órdenes solicitadas son atendidas con mayor velocidad.• Se puede saber de manera instantánea si se ha agotado algún platillo y así hacerlo saber al cliente, evitando la molesta notificación pasados unos minutos.• Se agiliza el proceso de pago de la cuenta.• Se protege el patrimonio y confianza del cliente ya que nunca pierde de vista su tarjeta bancaria, eliminando la posibilidad de que sea clonada.
Control de procesos	<ul style="list-style-type: none">• Se agiliza el proceso de atención al cliente, lo que reduce el tiempo de espera, mejora la satisfacción y la confianza en el negocio..• El registro de las acciones realizadas como platillos ordenados, hora de entrada y salida, mesero que atendió y día que se atendió, permitirá un análisis más profundo del comportamiento del negocio, generando información que sirva para optimizar y mejorar los procesos.• Se puede conocer cuales son los consumos de los platillos más solicitados y los que menos se piden de acuerdo a criterios como hora del día, temporada del año, fines de semana, etc.• Se tiene un mayor control del personal, ya que las acciones como entrada y salida del sistema, mesas atendidas, tiempo de servicio, etc. son registradas por el servidor, pudiendo implementar programas de estímulos, calificación y/o capacitación a los empeados.• Los cobros se generan automáticamente y sólo la persona asignada tiene contacto con el efectivo.

Ventaja	Descripción
Mejora administrativa	<ul style="list-style-type: none"> • La cuenta es generada automáticamente y almacenada en la base de datos, con lo que se cobra de manera exacta el consumo detallado. • Gracias a la arquitectura del Sistema, se pueden implementar fácilmente mecanismos para el control automático de la contabilidad que puede ser por día, semana, mes, año, etc., lo que ahorra costos ya que no es requerido tanto tiempo para esta actividad. Inclusive, con un ajuste específico a las necesidades del negocio, podría llegar a requerirse en un mínimo de los servicios de un contador. • El inventario se puede estimar con los datos manejados por el sistema, lo que permitiría avisar oportunamente a cocina de la probable falta de insumos.
Oportunidades de Negocios	<ul style="list-style-type: none"> • Aprovechando las bondades de los Web Services, se puede crear una relación Bussines to Bussines con un proveedor, de manera que cuando el sistema detecte la escasez de un insumo en el inventario, este puede solicitar una confirmación al área de compras (o insumos, o inventarios) para generar y enviar automáticamente la orden al proveedor. • El sistema SIR puede crecer con el negocio, facilitaría la creación de, por ejemplo, un sitio de servicio a domicilio o uno de banquetes a través de Internet y puede ser empleado por nuevas sucursales. • La información de todas las sucursales podría estar concentrada en una sola base de datos central, ayudando a la administración de la cadena del negocio. • Evitando la clonación de tarjetas se protege el negocio, al no perder la confianza y preferencia del cliente, a la vez que se colabora en la erradicación de un delito que ocasiona pérdidas por más de 700 millones de pesos al año tan sólo en México.
Tecnológicas	<ul style="list-style-type: none"> • Debido a que se trata de un Web Service, toda comunicación se hace a través de un protocolo estándar, lo que permite que los clientes sean programados en cualquier lenguaje con una API que soporte dicho protocolo (Java, Visual Basic, C#, etc.). • Los clientes inalámbricos pueden ser dispositivos PALM, WindowsCE o inclusive clientes Web, lo que el negocio decida que es mejor para sus necesidades. • El sistema SIR no está casado con ninguna tecnología de comunicación y se puede elegir de entre varios protocolos de comunicación inalámbrica presente o futura.

Además, a diferencia de otras soluciones similares existentes en el mercado, el sistema SIR es modular, permitiendo así ajustar el tamaño, requerimientos y costos de implementación a las necesidades específicas del usuario. Además, SIR está programado en Java, lo cual brinda una independencia de plataforma que otro tipo de sistemas no (están programados para un sistema operativo específico).

Desarrollo

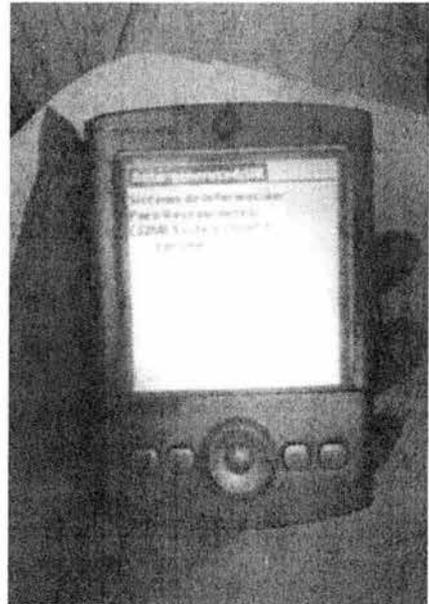
Fotografías del sistema SIR corriendo en distintos dispositivos y sistemas operativos:



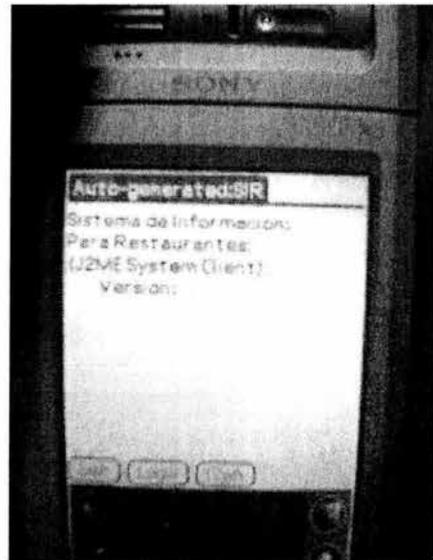
SIR en PalmOS Emulator (PalmOS 3.1, Palm m505)



SIR en un teléfono celular Sony-Ericsson modelo T610



SIR en una Palm modelo m550 (Tungsten T) con PalmOS 5.0



SIR en una Sony Clié modelo PEG-NX70V con PalmOS 5.0

Costos

Debido a las muchas posibilidades que brinda el sistema SIR para su producción, es difícil el mencionar cual sería el costo estimado de implementación ya que dependerá mucho del tipo y tamaño de negocio, así como sus preferencias de arquitectura.

Dicho lo anterior, podemos mencionar que el sistema SIR es mucho muy flexible en su costo final de implementación, dependiendo de las necesidades de cada uno de los negocios de Restaurantes. Se puede tratar de un negocio que esté conformado por sólo 5 meseros y que sea de espacios reducidos, y podría tratarse de una cadena de restaurantes con una gran cantidad de empleados, que ha decidido entrar al negocio por Internet.

La infraestructura más barata para desarrollar un sistema SIR podría constar de lo siguiente (los precios están dados en pesos y están estimados a la primer quincena del mes de Mayo de 2004):

- Equipos Palm con capacidades J2ME y Bluetooth (\$3,000 por mesero)
- Acces Point para la comunicación inalámbrica (\$3,000 por cada 7 meseros)
- Computadora Personal con antena bluetooth (\$15,000 Servidor)
- Computadora Personal con antena bluetooth (\$6,000 Cliente de caja y cocina)
- Sistema Operativo Linux (sin costo)
- Manejador de Base de Datos PostgreSQL o MySQL (sin costo)
- Servidor de aplicaciones JBOSS (sin costo)
- Costo del Sistema SIR (ver tabla en la siguiente página)

Inclusive, siendo estrictos, el acceso inalámbrico a redes es necesario únicamente para los clientes de los meseros, de forma que se pueden abatir un poco más los costos si los clientes de caja y cocina se conectan a una red alamburada (que bien puede ya existir). De esta forma, por una configuración básica, para un negocio pequeño de 7 meseros, se estaría haciendo una inversión en infraestructura que puede ir desde los \$38,000.00 pesos aproximadamente.

Añadiendo el costo de la aplicación (paquete básico) y su implementación, se podría hablar de un costo total de aprox. \$60,000.00 (sesenta mil pesos), lo cual es una inversión relativamente pequeña lograda gracias a que SIR es un sistema modular, extensible, independiente de plataforma, y basado en estándares abiertos. A continuación se presenta un esquema propuesto del precio de licencias del sistema SIR.

Las cantidades especificadas de los componentes indican el número de equipos o dispositivos en que se tiene derecho a instalar el sistema por el pago de licencia.

Lista de Precios (por componente por dispositivo)	
Componente	Precio (M.N.)
Servidor SIR (SS)	\$ 6,400.00
Cliente Mesero (CM)	\$ 2,880.00
Cliente Caja (CC)	\$ 995.00
Cliente Cocina (CK)	\$ 995.00

Precio por Paquetes de Componentes								
Cantidad Componente	Precio por Cantidad de dispositivos (M.N.)							
	<10	<20	<30	<50	<100	<500	<1000	+1000
CM	\$ 5,000	\$ 6,500	\$ 8,000	\$ 9,000	\$ 12,000	\$ 20,000	\$ 30,000	---
CC	\$ 2,000	\$ 3,000	\$ 4,000	\$ 5,000	\$ 6,500	\$ 9,000	\$ 12,000	---
CK	\$ 2,000	\$ 3,000	\$ 4,000	\$ 5,000	\$ 6,500	\$ 9,000	\$ 12,000	---
Cantidad Componente	Precio por Cantidad de dispositivos (M.N.)							
	1	2	3	5	10	30	50	100
SS	\$6,400	\$10,000	\$15,000	\$23,000	\$40,000	\$80,000	\$90,000	\$150,000

Precio por Paquetes Completos (por cantidad de dispositivos)					
Paquete	SS	CM	CC	CK	Precio (M.N.)
Simple	1	10	2	2	\$ 12,000.00
Básico	2	25	4	4	\$ 18,000.00
Medio	3	30	6	6	\$ 22,500.00
Grande	5	50	10	10	\$ 32,000.00
Extendido	7	60	15	15	\$ 39,000.00
Cadena 1	10	100	20	20	\$ 52,000.00
Cadena 2	30	250	60	60	\$ 97,000.00
Personalizado	---	---	---	---	---

Se ofrece al cliente la alternativa de armar su propio paquete de acuerdo a sus necesidades específicas, obteniendo un sistema y un costo personalizados.

El pago de la licencia da al cliente derecho a soporte técnico y actualizaciones del software por 1 año. El pago de licencia no incluye asistencia técnica para configuración del servidor, planeación y/o implementación de la infraestructura. El desarrollo de módulos adicionales no considerados en esta tabla (cliente de órdenes en línea por internet, cliente de banquetes, etc.) tienen un costo adicional.

5.

Conclusiones

5.1. Redes inalámbricas

En los últimos 2 años se ha visto un enorme crecimiento de las redes inalámbricas, que han pasado de ser algo costoso de uso en grandes corporaciones a soluciones completas para redes caseras y de oficinas de bajo costo. En México ya existen las redes inalámbricas públicas en aeropuertos, restaurantes y centros comerciales, impulsando el desarrollo de una red inalámbrica de cobertura total.

Las redes inalámbricas han evolucionado y mejorado el desempeño de las transmisiones, y hoy día se puede tener una red de 54Mbps con transmisión segura a distancias de hasta 100 metros, según las especificaciones mostradas por los fabricantes. Sin embargo, en un ambiente de trabajo real a esa distancia la velocidad de transmisión se reduce drásticamente; idealmente esa distancia no debería ser más de 40 metros. También se han desarrollado diferentes protocolos y estándares para la transmisión de datos sin cables y entre ellas podemos mencionar: Bluetooth 1.0, Bluetooth 1.2, IrDA, 802.11a, 802.11b, 802.11g, 802.16, GSM, WUSB.

Las grandes compañías de telefonía celular a nivel mundial están muy interesadas en desarrollar redes de transmisión de datos a través de sus celulares y para ello ya han creado estándares de comunicación como GSM y 3G, siendo 3G la última generación de redes con una cobertura a nivel mundial y velocidades de transmisión de hasta 10Mbps. Se espera que cuando 3G se adopte completamente (dentro de algunos años) cambiará radicalmente la forma en que nos comunicamos.

WUSB es una iniciativa de Intel Corp. que propone generar un nuevo estándar de transmisión de datos inalámbrica basado en el ya popular USB. La idea es tomar las ventajas que brinda la conexión USB 2.0 y trasladarla a una transmisión sin cables que logre tasas de transferencia de 200Mbps a distancias de hasta 100 metros. Por el momento está en fase de investigación y desarrollo, por lo que habrá que esperar, pero sin duda la tecnología es muy prometedora.

Bluetooth surgió como una solución de bajo costo y poco consumo de energía para redes que permitieran la comunicación entre dispositivos de cómputo a distancias cortas. Por tal motivo se pensó en su empleo para la comunicación entre periféricos de cómputo como impresoras, teclados y ratones, así como equipos portátiles (laptop) y PDA's. Sin embargo, esta tecnología no ha tenido la adopción que se esperaba y no ha cumplido con las altas expectativas que se crearon inicialmente. Velocidades de transferencia bajas (2Mbps) en comparación con otros protocolos, rangos de distancia máxima de 10 metros y problemas de compatibilidad entre dispositivos han ocasionado el poco éxito de Bluetooth. Se puede anticipar que Bluetooth se limitará a la conexión de dispositivos de uso personal.

WIFI es un conjunto de protocolos conformados por la familia IEEE 802 que tiene como objetivo desarrollar tecnologías de comunicación sin cables con un amplio rango de usos que van desde redes corporativas a redes caseras. Existen algunas variantes en el estándar pero en los últimos meses la versión que más ha sobresalido y se postula como la ganadora es 802.11g, que puede transmitir a velocidades de 54Mbps a una distancia de hasta 100 metros, y además es compatible con los estándares más viejos de la familia 802.11b (aunque en estos últimos, la tasa de transferencia máxima de 54Mbps no se puede conservar en distancias mayores a 50 metros).

5. Conclusiones

Otra de las recién creadas variantes es la 802.16 que si bien aún no está en fase de producción y deberemos esperarla un poco más, nos promete velocidades de transmisión de 100Mbps en rangos de distancias expresados en kilómetros, lo que la hace ideal para desarrollar Wireless MAN's (Metropolitan Area Networks).

Hoy día la mejor opción para desarrollar una red inalámbrica local y cumplir satisfactoriamente los requisitos de comunicación del sistema SIR es sin duda Wi-Fi en su variante 802.11g, que permite implantar una red de alta velocidad y buen rango de distancia con una baja inversión, de hecho es más barato el emplear una red inalámbrica que una tradicional ya que se evitan los gastos de instalación y alambrado, que suelen ser altos.

5.2. Servicios Web

¿Cual es la realidad de los Servicios Web?

No cabe duda que los Servicios Web son la tecnología más comentada hoy día por los profesionales de TI y se puede decir que existe una publicidad exagerada alrededor del tema. Existen dos opiniones muy contrastantes al respecto: un cierto grupo de profesionales por un lado ven a los servicios Web como la tecnología que transformará la forma de desarrollar y desplegar aplicaciones para las futuras generaciones. Por el otro lado, muchas personas ven a los servicios Web como simples programas basados en aceite resbaladizo que no tienen la oportunidad de lograr ningún impacto en las aplicaciones del mundo. La única manera justa de dimensionar el impacto potencial de los servicios Web es descubrir quien los está desarrollando, desplegando, o planeando para la industria. Sobre este caso, la firma de investigación Gartner (www.gartner.com) realizó dos investigaciones en el 2003 para medir la presencia de los servicios Web ahora y en un futuro próximo.

La primera investigación fue realizada con 50 empresas de Estados Unidos que tienen utilidades de más de \$500 millones de dólares. La segunda investigación alcanzó más de 100 empresas en Estados Unidos que tenían más de 1,000 empleados y que planeaban desplegar Servicios Web en los próximos 12 a 24 meses.

El gran proyecto para los Servicios Web es el de proveer un "framework" para desplegar aplicaciones tanto dentro de una empresa como fuera de ella. El despliegue de la aplicación puede ser construido con componentes de nuevo software escrito con estándares de servicios Web.

Los resultados de las investigaciones sugieren que la gran mayoría de las empresas (poco más del 85%) primero desplegarán los servicios Web para la integración de aplicaciones internas y nuevas aplicaciones que necesitan, y posteriormente desarrollarán Servicios Web fuera de la empresa. Otro resultado interesante, es que el 54% de las empresas involucradas en la investigación ya están trabajando en el desarrollo del servicios Web, ya sea en J2EE o en .NET.

Por otro lado, la competencia entre los distribuidores de soluciones apenas comienza; aún es temprano para declarar vencedores y perdedores. Sin embargo, las empresas deben elevar la relación con los distribuidores o hacer una nueva relación de negocios con la que consideren la mejor tecnología, y que sea financieramente viable. La investigación también indica que las empresas deberán estar preparadas para desplegar aplicaciones en cualquiera de las dos arquitecturas de despliegue de servicios web: tanto en J2EE como en .NET; no existe una sola arquitectura que lo comprenda todo.

¿Porqué no comenzar ahora?

Según lo encontrado en el estudio de Gartner, las grandes compañías están trabajando ya en la habilitación de XML de sus aplicaciones o lanzando nuevos proyectos para desarrollar Servicios Web. Aquellas compañías que no lo están haciendo tienen una actitud conservadora hacia la nueva tecnología, o alguien en la organización de TI está dormido en el mando. Hoy es un buen momento para despertar. Los Servicios Web están creciendo, y se conectarán con el mundo tarde o temprano. Entonces, ¿para que esperar?

5. Conclusiones

5.3. J2ME

Las Necesidades

En la actualidad, como se ha venido mencionando, existe una gran diversidad de soluciones de cómputo para el procesamiento de la información; una diversidad que se manifiesta desde los componentes más básicos de una computadora (hardware) hasta los más altos niveles de abstracción (Software). En los últimos años esta amplia diversidad en los sistemas ocasiona muy altos y dolorosos costos a las empresas que han querido integrar sus aplicaciones. Esto ocasiona en el mercado la necesidad de una *estandarización* de los métodos, medios y herramientas por los cuales se desarrollan y despliegan las aplicaciones de los negocios.

Como respuesta a la necesidad de un lenguaje que permita desarrollar rápida y sencillamente aplicaciones multiplataforma, confiables, robustas, flexibles, a bajo costo y reutilizables, se introduce Java al mercado; cuando se extiende la necesidad de disponer de esta información a los dispositivos portátiles, surge Java 2 Micro Edition (J2ME).

¿Porqué usar Java?

En el mercado de lenguajes de programación para dispositivos portátiles existen otras soluciones, en cierto sentido "mejores" que J2ME, pero conllevan ciertas desventajas. Pongamos el ejemplo del lenguaje de programación "C" para PalmOS en forma de tabla comparativa entre los lenguajes:

Comparación de Características y Ventajas de lenguajes de programación "C" vs. "Java"	
C	Java
No cuenta con un sistema para desarrollar GUI's.	Cuenta con AWT y Swing para desarrollar GUI's.
No se obliga al usuario a manejar errores.	Se proporciona un mecanismo de manejo de errores, y se obliga al usuario a manejar algunos de ellos.
Uso de librerías específicas del sistema, requiere mucha programación explícita.	Uso de las librerías de Java. Existe mucha funcionalidad implementada en librerías estándar.
Manejo explícito de recursos de bajo nivel (punteros, dispositivos y administración de memoria).	Manejo automático de punteros, referencias y liberación de memoria (<i>Garbage Collection</i>).
Manejo explícito de arreglos en memoria.	Los arreglos se crean como objetos.
Las cadenas se manejan como arreglos de caracteres.	Existen clases para el manejo de cadenas.
Modificado para brindar soporte a programación orientada a objetos.	Nació como un lenguaje completamente orientado a objetos.

La sintaxis de Java, así como sus operadores, son muy similares a los de C, así que no es tan difícil que un desarrollador que conozca C, aprenda rápidamente Java. En la práctica, Java puede tener sus desventajas, específicamente de desempeño, pero es un problema que se ha atacado conforme pasa el tiempo: se han hecho implementaciones *HotSpot*, compiladores *Just In Time*, además de existir numerosas herramientas y *profilers* que nos ayudan a optimizar los programas escritos en Java. Ciertamente, Java es un lenguaje de programación inmaduro si lo comparamos con lenguajes como C, pero el entusiasmo de los desarrolladores y las empresas lo han hecho evolucionar rápidamente, volviéndolo muy popular y ampliamente usado.

Siendo una plataforma estable, el futuro es muy promisorio para Java, y aunque todavía tiene que madurar, es destacable su diseño desde su propia concepción. Las empresas están utilizando Java como solución a problemas reales y de misión crítica; existen cientos de compañías que brindan soluciones muy robustas basadas en Java; las empresas reducen los costos de diseño, desarrollo e implementación de sus nuevos sistemas y facilitan la integración de los ya existentes.

Es un hecho que existe todo un mercado ya establecido y consolidado que seguirá creciendo mientras no aparezcan lenguajes de siguiente generación. Java seguirá desarrollándose y madurando gracias a su independencia de plataforma, flexibilidad y extensibilidad, y seguramente será, junto con XML, la piedra angular para el desarrollo y despliegue de las aplicaciones del futuro.

J2ME, ¿una alternativa real?

En cuanto a su uso en dispositivos portátiles, J2ME es una tecnología de reciente creación que aún no está bien probada. Los desarrolladores no tienen experiencia y se enfrentan a problemas muy diferentes a los que implica el desarrollo en servidores o computadoras de escritorio. El desempeño es algo en lo que aún se tiene mucho trabajo que hacer; la funcionalidad del lenguaje todavía es muy limitada, pero se han dado pasos importantes con la liberación de la especificación MIDP 2.0. Conforme avanza la tecnología, los dispositivos portátiles son más potentes, con más memoria y más compactos, al mismo tiempo que las implementaciones de Java utilizan mejor la potencia, memoria y capacidades nativas de los dispositivos.

Con todo el trabajo que se está invirtiendo sobre J2ME, en el corto plazo se convertirá en una plataforma más en la cual se puedan desarrollar aplicaciones, pero seguirá brindando ventajas que ningún otro lenguaje puede ofrecer. Al ser (igual que sus contrapartes, J2SE y J2EE) independiente de plataforma, será el lenguaje preferido para desarrollar aplicaciones de bajo costo, pudiendo tener opciones al elegir el dispositivo donde ejecutarlas.

En conclusión, parece inevitable que J2ME se vuelva parte importante dentro del mercado de dispositivos portátiles, y en conjunto con las tecnologías de redes inalámbricas (Bluetooth, Wi-Fi, WUSB, etc.), definitivamente determinarán el futuro en la forma de diseñar, desarrollar y desplegar aplicaciones móviles.

5. Conclusiones

5.4. Conclusiones Finales

Finalmente, cabe destacar que todos los esfuerzos por la estandarización, por el desarrollo de las redes inalámbricas, por el desarrollo de Java y por la adopción de los Servicios Web, están orientados siempre a reducir los costos de operación de las empresas, y de ahí se derivan varios aspectos: la competitividad, la eficiencia de los procesos, la disponibilidad de la información estratégica, etc. Una tecnología que brinde a las empresas estos beneficios, será siempre bien recibida, pero hay que tomar en cuenta que ninguna empresa estaría dispuesta a invertir en una nueva tecnología cada 6 meses o cada año. Las tecnologías se deben desarrollar sobre bases sólidas, bien pensadas y que comprueben funcionar correctamente, de lo contrario se pueden convertir en un riesgo de inversión para los negocios.

Las redes inalámbricas, J2ME y los servicios Web son tecnologías de muy reciente creación, su adopción ha sido un poco lenta y cautelosa, y todo ello pudiera parecer un riesgo de inversión. Es cierto que aún hay trabajo que hacer en cada una de ellas, pero si hay algo que el tiempo ha comprobado, es que un negocio demasiado conservador, que no se involucra con los avances tecnológicos, está automáticamente condenado a convertirse en poco competitivo, poco eficiente, poco viable y finalmente al fracaso.

Las tecnologías de Servicios Web parecen ser la evolución natural en el diseño de sistemas como lo ha sido la programación orientada a objetos a los lenguajes de programación. Aunque en un principio los diseñadores de sistemas tomaron con cautela el uso de lenguajes orientados a objetos, éstos han probado ser un modelo robusto que se usa en la actualidad en los sistemas más demandantes de misión crítica, y se prevee que suceda de la misma forma con el modelo de sistemas basados en componentes hechos con tecnologías y protocolos estándares que proporcionan los Servicios Web.

Las empresas deben comenzar a prepararse hoy mismo para enfrentar los retos de un mundo globalizado, en donde para ser competitivos, la información tendrá que estar disponible *"a quien sea, a cualquier hora, en cualquier lugar, en cualquier dispositivo"*.

6.

Anexos

*** GLOSARIO**

- ▷ **802.11.** Tecnología de transmisión de datos por medios inalámbricos desarrollada y promovida por la Wi-Fi Alliance. Wi-Fi provee una red de datos inalámbrica de un alcance de hasta 100 metros a velocidades de entre 11Mbps y 54Mbps
- ▷ **A posteriori.** Esta palabra es un calificativo para la forma de obtener conocimiento. Un conocimiento es "a posteriori" cuando no se puede obtener sin una experiencia o conocimientos previos.
- ▷ **A priori.** Esta palabra es un calificativo para la forma de obtener conocimiento. Un conocimiento es "a priori" cuando se puede obtener independientemente de la experiencia o como inferencia o deducción de conocimientos previamente adquiridos.
- ▷ **Access Point.** Ver *Punto de Acceso*.
- ▷ **Activos Digitales.** Se refiere a los activos del negocio almacenados en forma digital (por ejemplo: una base de datos con información estratégica, reportes financieros, datos de los clientes y proveedores, etc.).
- ▷ **ADO.** ActiveX Data Objects. Es el componente de acceso a fuentes de datos de lenguaje neutral de Microsoft.
- ▷ **ADO.NET.** ActiveX Data Objects .NET. Es el componente de acceso a fuentes de datos de lenguaje neutral de la plataforma .NET de Microsoft. Es la contraparte de JDBC para componentes J2EE.
- ▷ **Ambiente Gráfico.** Se refiere a los recursos que proporciona un Sistema Operativo para dibujar elementos gráficos en la pantalla (ventanas, botones, menús, íconos, imágenes, video, etc.) y permitir una mejor y más sencilla interacción con el usuario.
- ▷ **Amplificación de una señal.** Un amplificador es un sistema que tiene a su salida una réplica de la señal de entrada, cuya amplitud fue aumentada. Generalmente se usa para compensar pérdidas por atenuación de la señal en un canal de transmisión.
- ▷ **Ancho de Banda.** El ancho de banda en telecomunicaciones se refiere a un rango de frecuencias dentro del cual se puede transmitir información en un canal. Mientras más ancho sea este rango de frecuencias, existe la posibilidad de transmitir a una mayor velocidad.
- ▷ **ANSI.** American National Standards Institute. Organismo internacional encargado de desarrollar y proclamar estándares.
- ▷ **API.** Application Programming Interface. Una API está conformada por una o varias librerías en algún lenguaje de programación específico que nos dan alguna funcionalidad que podemos integrar en alguna otra aplicación. Generalmente, dentro de la API se incluye la documentación de toda la funcionalidad que tiene implementada.

6. Anexos

- ▷ **Applet.** Tecnología de Java que permite correr aplicaciones descargadas de Internet dentro de un entorno de ejecución compatible, como AppletViewer, Internet Explorer o Netscape.
- ▷ **AppleTalk.** Una arquitectura de redes LAN de bajo costo desarrollada por Apple Computer Inc. Las redes que utilizan AppleTalk, también son conocidas como redes LocalTalk.
- ▷ **Árbol de Análisis Sintáctico.** Es una representación jerárquica de las clases gramáticas de un lenguaje. Permiten verificar la correcta formación de las frases y son empleados para traducir de un lenguaje a otro. En lenguajes de programación, se utilizan para realizar la depuración de programas y en ocasiones para reconstruir un código fuente a partir de código objeto.
- ▷ **ARPANET.** Red de comunicaciones de aplicación militar que fue financiada a partir de 1968 por la ARPA (Advanced Research Projects Agency, Agencia de Proyectos de Investigación Avanzada) del pentágono en E.U., que a la postre daría origen a lo que hoy conocemos como Internet.
- ▷ **ASP.** Active Server Pages. Tecnología server-side de Microsoft, que permite tener en un archivo mezclado código HTML y Visual Basic, compilarlo y publicarlo en un servidor Web para generar contenido dinámico.
- ▷ **AWT.** Abstract Window Toolkit. Paquete de API's de Java que sirven para generar GUI's (Graphic User Interface, Interfaz Gráfica de Usuario).
- ▷ **B2B.** Bussines to Bussines. Es la forma de conocer la relación e interacción que existe entre una empresa y sus socios y proveedores
- ▷ **B2C.** Business to Consumer. Es la forma de conocer la relación e interacción que existe entre una empresa y sus clientes
- ▷ **BigEndian.** Tipo de Codificación de números binarios, en donde la cifra más significativo del número se almacena en la menor dirección de memoria (es decir: en un acceso secuencial a memoria, el byte más significativo se almacena primero). Esto es contrario a la codificación LittleEndian. Así como existe codificación BigEndian y LittleEndian para los bytes, de igual forma aplica para almacenar bits en una localidad de memoria.
- ▷ **BizTalk.** Plataforma desarrollada por Microsoft para permitir la realización de transacciones entre sistemas implementados con .NET.
- ▷ **Bluetooth.** Es un consorcio formado por varias compañías líderes en el desarrollo de hardware y software de telecomunicaciones, llamado *Bluetooth SIG (Special Interest Group)* y cuya función es el desarrollo y promoción de una tecnología de transmisión de datos por medios inalámbricos, a una velocidad teórica de hasta 1 Mbps, con un alcance de hasta 10 metros. El SIG es una organización privada fundada en 1998, y con sede en Overland Park, Kansas, E.U, a partir de julio del 2002.

- ▷ **bps.** Acrónimo en inglés de “bits per second” (bits por segundo). Es una medida de velocidad de transmisión de datos digitales, y mide la cantidad de bits que pueden viajar a través de un canal de transmisión desde el emisor hasta el receptor en un segundo.
- ▷ **Bytecode.** Codificación que utiliza el lenguaje de programación Java para compilar sus programas. El bytecode es un código de instrucciones propio de Java e independiente de plataforma, lo cual le da al lenguaje sus características de portabilidad.
- ▷ **CAFEBABE.** La especificación de Java emitida por Sun, indica que toda clase de java en bytecode debe comenzar con el número 0xCAFEBABE hexadecimal (3405691582 decimal). Esto sirve para identificar si un archivo es en realidad un bytecode Java, y la razón de haber escogido ese número en particular parece provenir de la relación entre el café y la palabra “Java”.
- ▷ **Caja de Arena.** Es un mecanismo de seguridad que algunos lenguajes utilizan para proteger los recursos de un sistema. Cuando un programa corre en una sandbox (caja de arena), se le niega el acceso a cualesquiera recursos que no le estén explícitamente asignados dentro de su entorno de ejecución. Esto aplica especialmente a aplicaciones sobre las cuales no conocemos su origen ni podemos garantizar que no dañarán el sistema.
- ▷ **CDC.** Connected Device Configuration. Es una configuración que define el conjunto de librerías de J2ME para dispositivos inalámbricos.
- ▷ **CDMA.** Code Division Multiple Access. Tecnología de multiplexión desarrollada y usada durante la segunda guerra. CDMA permite combinar varios mensajes en uno sólo y usar todo el ancho de banda disponible para su transmisión. La codificación CDMA consiste en realizar ciertas operaciones sobre varios mensajes y convertirlo en uno sólo, mismo que mediante una operación inversa, es decodificado en sus mensajes originales por el receptor. La ventaja de este sistema es que se transmite en realidad un sólo mensaje por el canal, su desventaja es que el tamaño del mensaje combinado puede ser mucho más grande que la suma del tamaño de los mensajes originales.
- ▷ **CGI.** Common Gateway Interface. Es un estándar que especifica los mecanismos mediante los cuales se pueden comunicar un cliente web con un servidor web. Este mecanismo se utiliza en muchos lenguajes (como por ejemplo Perl, PHP, JSP y ASP) para generar contenido dinámico o enviar y recibir información por medio de páginas Web.
- ▷ **Cifrado Asimétrico.** Ver *Cifrado de Llave Pública*.
- ▷ **Cifrado de Llave Privada.** Tipo de encriptación en que la llave de cifrado y la de descifrado son la misma. Para descifrar el mensaje es necesario conocer la llave.
- ▷ **Cifrado de Llave Pública.** Tipo de encriptación en que la llave de cifrado es diferente a la llave de descifrado. Para descifrar el mensaje, es necesario conocer la llave pública y la llave privada con las cuales fue cifrado, por ello es mucho más seguro que el cifrado por llave privada.

6. Anexos

- ▷ **Cifrado simétrico.** Ver *Cifrado de Llave Privada*.
- ▷ **Cifrado.** Técnica criptográfica por la cual un mensaje inteligible se transforma en otro mensaje, en función de una llave de cifrado, volviéndolo ininteligible.
- ▷ **Circuitos de Comunicación.** Es el conjunto de recursos que se utilizan para establecer una comunicación desde un emisor hasta un receptor (canal(es) de comunicación, ancho(s) de banda, medio(s) de transmisión, codificador(es), decodificador(es), etc.).
- ▷ **ClassLoader.** Es la clase en Java encargada de leer los archivos de programas java en bytecode, verificarla y cargarla para su ejecución dentro del JRE.
- ▷ **CLDC.** Connected Limited Device Configuration. Es una configuración que define el conjunto de librerías de J2ME para dispositivos inalámbricos con poder de procesamiento limitado.
- ▷ **Client-side.** En un ambiente cliente-servidor es cuando una aplicación o un proceso es ejecutado en el dispositivo del cliente.
- ▷ **CLR.** Common Language Runtime. Es el entorno de ejecución soportado por .NET de Microsoft. Se puede ver como la contraparte del JRE. El CLR lee y ejecuta programas compilados en un código especial independiente de plataforma. El compilador para JRE soporta varios lenguajes, como C++, SmallTalk, C#, COBOL, Fortran, Pascal, Perl, Python, Visual Basic, entre otros.
- ▷ **Codificación de la fuente.** Es el proceso necesario para convertir una señal analógica en alguna codificación específica; consiste en la conexión en serie de un muestreador, un cuantizador y un codificador.
- ▷ **Codificación.** Se refiere a la representación de un determinado mensaje requerida para que éste pueda viajar por un medio de transmisión específico.
- ▷ **Código Objeto.** Es el código de instrucciones generado por algún compilador para ser ejecutado en alguna arquitectura. Casi siempre este código objeto requiere una etapa de adicional para poder ser ejecutado (ligado o linking).
- ▷ **COFETEL.** Comisión Federal de Telecomunicaciones. "La Comisión es un órgano administrativo desconcentrado de la Secretaría de Comunicaciones y Transportes, con autonomía técnica y operativa, el cual tendrá las atribuciones que le confiere el Decreto de Creación y el Reglamento Interior de la Secretaría de Comunicaciones y Transportes, con el objeto de regular y promover el desarrollo eficiente de las telecomunicaciones".⁵
- ▷ **COM.** Component Object Model. Es una serie de especificaciones y API's implementadas por Microsoft que permiten acceder a objetos y métodos dentro de un mismo sistema. Es la contraparte centralizada de DCOM.

⁵ COFETEL (http://www.cofetel.gob.mx/quienes_somos_desc.html)

- ▷ **Comunicación Asíncrona.** Este tipo de comunicación sucede cuando un emisor abre un canal de comunicación, envía un mensaje a un receptor y cierra la conexión sin esperar respuesta, la cual puede llegar posteriormente o no llegar nunca. Como ejemplo tenemos el protocolo UDP, que permite comunicaciones asíncronas a través de Internet.
- ▷ **Comunicación Síncrona.** Se refiere al hecho de que en un sistema, una entidad transmisora emite un mensaje y espera una respuesta proveniente del receptor sin abrir una nueva conexión. También se asume por regla general el hecho de que el sistema que emite el mensaje queda bloqueado (es decir: no sigue su flujo de ejecución) sino hasta el momento de recibir respuesta por parte del receptor, o la conexión cumpla su periodo de "timeout". Como ejemplo está el protocolo TCP, que asume una comunicación síncrona entre 2 sistemas.
- ▷ **Confusión (Criptografía).** Las características de confusión de un algoritmo de cifrado tienen la finalidad de disminuir (idealmente eliminar) la relación que pueda existir entre los símbolos de un mensaje claro y su correspondiente cifrado.
- ▷ **Conjunto de Instrucciones.** Se refiere a todas las instrucciones en formato binario que algún procesador en específico es capaz de entender. Dentro de la especificación de un conjunto de instrucciones, se debe indicar cual es el valor binario de cada instrucción, qué es lo que hace cada una de ellas, qué parámetros requiere y de qué forma son obtenidos. La incompatibilidad de un procesador a otro, se debe precisamente a que sus conjuntos de instrucciones son diferentes.
- ▷ **Conmutación.** Técnica por la cual se puede alternar la transferencia de datos de diferentes mensajes a través de un mismo medio de transmisión. En redes de datos, esta conmutación puede ser por paquetes (se transfieren alternadamente paquetes o "pedazos" de cada mensaje) o por circuitos (se reserva un canal físico para la transmisión exclusiva de un mensaje en particular).
- ▷ **Contenedor de Aplicaciones.** Es un software cuya función es proveer cierta funcionalidad y servicios (como el ciclo de vida de una aplicación, el control de sus transacciones, su persistencia, su seguridad, el balanceo de cargas, etc.) a una aplicación diseñada especialmente para correr dentro de un contenedor.
- ▷ **CORBA.** Common Object Request Broker Architecture. Es una arquitectura estándar desarrollada por el *OMG* que nos permite publicar aplicaciones de manera consistente a través de diferentes arquitecturas, de forma que el acceso a objetos y métodos es estándar, independientemente de que las aplicaciones estén en sistemas, arquitecturas e inclusive lenguajes de programación diferentes.
- ▷ **Cracker.** Persona que se dedica a vulnerar los mecanismos de seguridad de los sistemas (sistemas operativos, software de aplicación, protocolos, etc.), generalmente en beneficio propio o de terceros (como poder usar software pirata u obtener información confidencial) o simplemente con la intención de dañar los sistemas y entorpecer, o inclusive interrumpir su buen funcionamiento.

6. Anexos

- ▷ **CRC.** Cyclic Reduncy Check. Es una técnica que, a través de un algoritmo, permite la verificación de la integridad de un conjunto de datos y/o detectar errores de transmisión en redes de datos. CRC consta de información adicional agregada a un conjunto de datos cuyo valor está en función de los datos mismos. Cuando se requiere verificar la integridad de los datos, se hace un cálculo del CRC (en función del valor del mensaje), el cual tiene que coincidir con el CRC adjunto a los datos. En caso de que la cadena calculada no coincida con la adjunta, se asume que ha sucedido una corrupción en los datos, y típicamente se requiere de la sustitución de la información corrupta o la retransmisión del paquete de datos al emisor. Esto es debido a que CRC permite la detección, pero no la corrección de errores.
- ▷ **Criptoanálisis.** Ciencia que estudia la forma de descifrar un mensaje sin la necesidad de conocer a priori la llave y/o el algoritmo de descifrado.
- ▷ **Criptografía.** Ciencia que estudia la forma de cifrar un mensaje para protegerlo contra su visualización por destinatarios no autorizados, haciéndolo ininteligible a quien no posea información sobre la forma del descifrado.
- ▷ **Criptología.** Es la ciencia formada por la *Criptografía* y el *Criptoanálisis*.
- ▷ **Daemon (Informática).** Ver *Demonio*.
- ▷ **Datagrama.** Un datagrama es un paquete de datos que se envía en los protocolos no orientados a conexión. Un datagrama siempre es enviado por una red de datos sin esperar respuesta alguna del destinatario. El datagrama puede o no llegar a su destino, y el destinatario puede o no responder el mensaje. Este tipo de protocolos son útiles cuando se desea hacer mas rápida una comunicación (ya que no se tiene que esperar una respuesta), optimizar el uso del ancho de banda (ya que por la red no viajan paquetes de respuestas) y/o no es requisito indispensable recibir la totalidad de los paquetes en una comunicación, por ejemplo: en audio y video en demanda, o en juegos en red.
- ▷ **DCOM.** Distributed Component Object Model. Es una serie de especificaciones y API's implementadas por Microsoft que permiten acceder a objetos y métodos en sistemas remotos distribuidos.
- ▷ **Demonio (Informática).** Se conoce como *demonio* o *daemon* a un programa que está cargado en la memoria de una computadora esperando un cierto evento en el sistema para realizar alguna acción (por ejemplo: atender una petición de conexión, un periférico, emitir un mensaje, etc.).
- ▷ **DEMUX.** Es un dispositivo que tiene la tarea de separar un mensaje en sus mensajes individuales originales, a partir de uno previamente multiplexado por un MUX. CDMA y TDMA son ejemplos de métodos de multiplexión.
- ▷ **DES.** Data Encryption Standard. Estándar de cifrado de datos de llave simétrica, desarrollado en 1976 y proclamado estándar por la ANSI en 1981 (ANSI X.3.92). Utiliza una llave de 56 bits para cifrar por bloques de 64 bits de datos.
- ▷ **Descifrado.** Técnica por la cual un mensaje previamente cifrado, se transforme en el mensaje inteligible original mediante una llave de descifrado.

- ▷ **Difusión (Criptografía).** Las características de difusión de un algoritmo de cifrado buscan disminuir (idealmente eliminar) el efecto de redundancia de símbolos de un idioma al ser transformado en un texto cifrado.
- ▷ **Distorsión.** En comunicaciones, se refiere a toda modificación que puede tener una señal que llega a un receptor, con respecto a la señal original que envió el emisor.
- ▷ **DoS (Ataque).** Denial Of Service. Tipo de ataque mediante el cual un atacante manda peticiones a algún servicio en algún sistema hasta el punto de sobrepasar su capacidad y evitar la conexión de usuarios legítimos o incapacitar al sistema para atender más peticiones.
- ▷ **Duplex.** Modo de comunicación bidireccional. Es un modo de comunicación que puede enviar más de un mensaje simultáneamente en cualquier dirección, de modo que el emisor puede enviar un mensaje al mismo tiempo que recibe respuesta al uno que haya enviado anteriormente.
- ▷ **ebXML.** Electronic Business XML. Es un conjunto de especificaciones y estándares dictados por un consorcio de industrias. Estos estándares están encaminados a permitir realización de transacciones a través de Internet entre empresas de cualquier tamaño en cualquier lugar del mundo.
- ▷ **EJB.** Enterprise Java Beans. Es una tecnología Java que nos permite abstraer la información de Bases de Datos, directorios, archivos o cualquier otra fuente de datos, en forma de pequeñas aplicaciones o módulos, dándonos una vista y un mecanismo consistente de manejo de la información, independientemente de dónde resida físicamente.
- ▷ **ElGamal (Criptografía).** Técnica de cifrado de llave pública basado en el problema del logaritmo discreto.
- ▷ **End-to-end.** Final a final. Se refiere al modelo de un framework o arquitectura de información. Aplica a aquellas tecnologías que engloban la totalidad de las capas de un sistema (por ejemplo: una arquitectura que abarque soluciones desde la capa de almacenamiento con un *RDBMS*, hasta la capa de presentación de los clientes, e incluyendo todas las capas intermedias)
- ▷ **Entorno Gráfico.** Ver *Ambiente Gráfico*.
- ▷ **Error (Java).** En Java, se refiere a algún error que ocurre fuera del alcance de la aplicación y por tanto no es posible recuperarse de él (por ejemplo: una localidad de memoria inservible o un sector del disco duro dañado).
- ▷ **Espacio de Nombres.** En programación, es un espacio donde se almacenan los símbolos de un programa. Generalmente, los espacios de nombres no permiten la definición de símbolos duplicados, y por esta razón muchos lenguajes de programación (como Java) permiten la definición de múltiples espacios de nombres.
- ▷ **Estacionario (Satélite).** Ver *Geoestacionario*.

6. Anexos

- ▷ **Estructurada (Programación).** Es una forma de diseñar programas y lenguajes de programación. En una programación estructurada, el control del programa se hace por flujos, decisiones y bifurcaciones. Ejemplos de estos tipos de lenguajes pueden ser Pascal, CLIPPER, C, BASIC, etc.
- ▷ **Ethernet.** Es una arquitectura de redes LAN desarrollada por Xerox Corporation, en cooperación con DEC e Intel en 1976. Puede usar una topología de red de bus o estrella y actualmente alcanza velocidades de hasta 10Gbps. Es la base del estándar 802.3 de la IEEE.
- ▷ **Exception (Java).** En Java es una clase de objetos que indican que dentro de una aplicación ha ocurrido algún error que puede ser manejado, corregido o ignorado. Las Excepciones proveen los mecanismos que permiten la recuperación de errores dentro de programas hechos en Java.
- ▷ **Firewall.** Es un sistema implementado por hardware o software, cuya función es filtrar el tráfico desde Internet hacia alguna red interna y viceversa, basado en determinados criterios o reglas. Esto se hace principalmente por seguridad (evadir ataques de DoS, Virus, Intrusiones, etc.) y para optimizar el uso de los recursos de la red (evitar el mal uso del ancho de banda, tráfico ilegal en redes P2P, etc.).
- ▷ **Firma Digital.** Es un mecanismo que permite identificar a un objeto dentro de un sistema (un programa, una computadora o un usuario) por medio de una secuencia de datos única a ese objeto. La secuencia de datos que conforma la firma digital, puede extraerse de un mensaje y verificarse para garantizar que fue el objeto quien realmente emitió dicho mensaje.
- ▷ **Framework.** Entorno de Trabajo. Es la manera de conocer a una infraestructura específica de diseño, desarrollo y/o implementación de sistemas, incluyendo herramientas, software, protocolos, IDE's de desarrollo, plataformas, etc.
- ▷ **Garbage Collector.** Ver *Recolector de Basura*.
- ▷ **Gbps.** Acrónimo de Giga bps (o billones de bps)
- ▷ **GC.** Garbage Collector. Ver *Recolector de Basura*.
- ▷ **Geoestacionario (Satélite).** Término que se refiere al hecho de que un satélite está girando en órbita con la tierra de tal forma que se encuentra arriba de un mismo punto geográfico todo el tiempo.
- ▷ **GPRS.** General Packet Radio Service. Es un método de transmisión de información digital a través de una red de telefonía celular que brinda una velocidad de conexión de hasta 172kbps.
- ▷ **Graphic User Interface.** Ver *Interfaz Gráfica de Usuario*.
- ▷ **GSM.** Global System for Mobile communications. Es un sistema de comunicación para telefonía celular de banda angosta que utiliza multiplexión TDMA, y permite hasta 8 circuitos de comunicación sobre una misma frecuencia.
- ▷ **GUI.** Graphic User Interface. Ver *Interfaz Gráfica de Usuario*.

- ▷ **Gusano (Informática).** Es un tipo de *virus* de computadora cuya tarea es (además de infectar un sistema) buscar mecanismos para replicarse y distribuirse a sí mismo para tratar de infectar a más sistemas.
- ▷ **Hacker.** Se refiere a una persona muy especializada en alguna aplicación particular (un programa, un protocolo, una plataforma, un sistema operativo, una arquitectura, etc.) que por lo general conoce en gran parte los detalles de la implementación de esa aplicación (sus fortalezas, debilidades, bondades, vulnerabilidades, etc.).
- ▷ **Half Duplex.** Modo de comunicación bidireccional. En este modo de comunicación, existe un canal por donde puede viajar un solo mensaje a la vez, pero puede viajar en ambas direcciones, de forma que el emisor envía un mensaje al receptor, y mediante algún mecanismo de sincronización, en determinado momento los roles se invierten y se puede recibir una respuesta a través del mismo canal.
- ▷ **Heap.** Ver *Montículo*.
- ▷ **Hilo de Ejecución (Java).** Ver *Thread*.
- ▷ **HTML.** Hyper Text Markup Language. Lenguaje estándar basado en SGML cuya finalidad es describir la forma y la apariencia que tendrá la información en el momento de ser desplegada (generalmente dentro de un navegador Web).
- ▷ **HTTP.** Hyper Text Transfer Protocol. Es un protocolo estándar para la transmisión de información a través de una red de datos, aunque se concibió originalmente para permitir la transferencia de archivos de HTML.
- ▷ **HTTPS.** HTTP Seguro. Es el uso del protocolo HTTP por medio de un socket asegurado por SSL, es decir: el protocolo HTTP se mantiene inalterado, pero se "encapsula" dentro de un mensaje SSL, que está encargado de cifrar el mensaje HTTP, de forma que si un receptor distinto al destinatario intercepta el mensaje, lo recibirá encriptado. A este proceso de "encapsular" un mensaje en otro tipo de mensaje para su transmisión se denomina "tuneleo" (tunneling).
- ▷ **IDE.** Integrated Development Environment. Son aplicaciones diseñadas específicamente para desarrollar aplicaciones mediante una interfaz gráfica de usuario. Generalmente las IDE's proveen bondades como generación automática de código, revisión de sintaxis mientras se escribe, editores gráficos de GUI's, etc.
- ▷ **IEC.** International Electrotechnical Commission. Organismo internacional encargado de desarrollar y proclamar estándares sobre ingeniería eléctrica y electrónica.
- ▷ **IEEE.** Institute of Electric and Electronic Engineers. Organismo internacional encargado de desarrollar y proclamar estándares sobre ingeniería eléctrica y electrónica.
- ▷ **IETF.** Internet Engineering Task Force. Organismo Internacional encargado de desarrollar y proclamar estándares referentes a Internet.

6. Anexos

- ▷ **Infinito (Euclides).** Concepto matemático establecido a partir del quinto postulado de la geometría Euclidiana (por un punto fuera de una recta sólo se puede trazar una paralela a esta última). En términos generales, este postulado implica la complicada idea de que dos líneas paralelas se intersectan en el infinito (concepto contrario a la propia definición de líneas paralelas).
- ▷ **Infraestructura.** La infraestructura de algún sistema, cualquiera que sea su tipo, se refiere a las provisiones, mecanismos, herramientas, y recursos con que se cuentan para su implementación.
- ▷ **Instruction Set.** Ver *Conjunto de Instrucciones*.
- ▷ **Interfaz Gráfica de Usuario.** Se refiere a la forma de desplegar una aplicación a un usuario en un ambiente gráfico de forma que sea más amigable y más fácil de usar, a través del uso de elementos como ventanas, íconos, imágenes, botones y menús que pueden ser manipulados por dispositivos adicionales al teclado, como un mouse o un touchscreen.
- ▷ **IPv4.** Internet Protocol versión 4. Es el protocolo estándar que se utiliza para la comunicación entre sistemas a través de Internet.
- ▷ **IPv6.** Internet Protocol versión 6. Es el protocolo estándar que se utiliza para la comunicación entre sistemas a través de una red Internet 2.
- ▷ **IPX.** Internet Packet eXchange. Es un protocolo de comunicación de red basado en datagramas usado por el sistema operativo Novell NetWare.
- ▷ **IrDA.** Infrared Data Association. Consorcio de fabricantes de dispositivos encargado de desarrollar y promover la tecnología de transmisión de datos a través de ondas de luz dentro del espectro del infra rojo. Esto permite la comunicación entre 2 dispositivos sin la necesidad de cables. IrDA es un medio de transmisión de muy bajo consumo de potencia y un alcance medio. Su desventaja es que para poder establecer el circuito de comunicación, debe existir una línea visual entre los elementos emisor y receptor de ambos dispositivos.
- ▷ **ISO.** International Standardization Organization. La organización Internacional para la estandarización es la institución de desarrollo y proclamación de estándares más importante a nivel mundial.
- ▷ **ITU.** International Telecommunications Union. Organismo internacional encargado de desarrollar y proclamar estándares sobre telecomunicaciones.
- ▷ **J2EE.** Java 2 Enterprise Edition. Se refiere al conjunto de librerías extendidas de Java orientadas al desarrollo de aplicaciones empresariales. J2EE incluye librerías de soporte para JPS, EJB, Servlets, entre otras.
- ▷ **J2ME.** Java 2 Micro Edition. Es el conjunto de librerías de Java orientadas a su uso en dispositivos portátiles, como celulares y PDA's.
- ▷ **J2SE.** Java 2 Standard Edition. Se refiere al conjunto de librerías de Java estándar.

- ▷ **Java.** Lenguaje de programación orientado a objetos desarrollado por James Gosling en la década de los 90 para Sun Microsystems.
- ▷ **JDBC.** Java DataBase Connectivity. Es una especificación de Java que permite tener una API consistente para el acceso a fuentes de datos independientemente de dónde residan físicamente. Es la contraparte de ADO para componentes .NET.
- ▷ **JIT (Compilador).** Compilador Just In Time. Es una tecnología que utilizan las JVM que permite compilar bytecode de java y transformarlo al conjunto de instrucciones nativo de la arquitectura en que está corriendo la máquina virtual. Esto elimina el costoso proceso de interpretación de bytecode en los programas Java, brindando un mucho mejor desempeño.
- ▷ **JRE.** Java Runtime Environment. Todo programa hecho en java requiere forzosamente de un entorno de ejecución para correr, pero tomando en cuenta que Java tiene su propio conjunto de instrucciones codificadas en bytecode, es forzoso contar con un ambiente de ejecución que lo comprenda. Este ambiente (que brinda la máquina virtual de java), se denomina Java Runtime Environment (Entorno de Ejecución de Java).
- ▷ **JSP.** Java Server Pages. Es una tecnología Java que nos permite generar contenido dinámico en Internet, mediante aplicaciones server-side, en un archivo que permite mezclar lenguaje Java y HTML, compilarlo y publicarlo en un servidor Web.
- ▷ **JVM (Java Virtual Machine).** Programa que corre en alguna arquitectura en específico, y cuya función es interpretar los programas hechos en java que son leídos en forma de archivos de bytecode y traducirlos a código nativo de la arquitectura para que puedan ser ejecutados. La JVM es la base de la portabilidad de los programas hechos en Java.
- ▷ **kbps.** Acrónimo de kilo bps (miles de bps)
- ▷ **KVM.** K Virtual Machine. Es el nombre dado a la JVM de J2ME. El prefijo *k* se aplica debido a que ésta máquina virtual ocupa sólo unos cuantos kilobytes de almacenamiento en el dispositivo.
- ▷ **LAN.** Local Area Network. Se refiere a toda la infraestructura que conforma una red de transmisión de datos en un área relativamente pequeña (dentro del hogar o dentro de la oficina).
- ▷ **Latencia.** Por latencia, se entiende un retardo en la respuesta de un sistema entre el momento de la señal de entrada y la de salida. Existen muchos sistemas que presentan latencia, como una memoria RAM, una conexión a red, una tarjeta de video, una tarjeta de sonido o un disco duro .
- ▷ **Ligado.** Etapa en la que un programa en código objeto se relaciona (o liga) con librerías y demás símbolos externos para poder ser ejecutado por el sistema operativo.
- ▷ **Linking.** Ver *Ligado*.

6. Anexos

- ▷ **LittleEndian.** Tipo de Codificación de números binarios, en donde la cifra menos significativo del número se almacena en la menor dirección de memoria (es decir: en un acceso secuencial a memoria, el byte menos significativo se almacena primero). Esto es contrario a la codificación BigEndian. Así como existe codificación LittleEndian y BigEndian para los bytes, de igual forma aplica para almacenar bits en una localidad de memoria.
- ▷ **Logaritmo Discreto (Problema).** Se trata de un problema que debido a que requiere de mucho poder y tiempo de cómputo para ser resuelto, se utiliza en algoritmos de cifrado. El hecho es que, a pesar de parecer sencillo, no existe una forma eficiente de encontrar un número n tal que $y=x^n$, sabiendo que x y y pertenecen a un grupo finito.
- ▷ **LUCIFER (Criptografía).** Es una técnica de cifrado de bloques desarrollada por Horst Feistel para IBM en la década de los 60. LUCIFER es el precursor directo del estándar DES.
- ▷ **Marcar e Intercambiar.** Técnica por la cual el Recolector de Basura de Java mantiene una lista de objetos referenciados y candidatos a recolección. Se basa en marcar periódicamente con un bit los objetos que son referenciados por el programa para pasarlos a un "montículo". Cuando un objeto o un montículo quedan sin referenciar, automáticamente son candidatos a recolección. Esta técnica resulta útil cuando se pueden llegar a tener estructuras de datos circulares.
- ▷ **Mark and Sweep.** Ver *Marcar e Intercambiar*.
- ▷ **Mbps.** Acrónimo de Mega bps (Millones de bps).
- ▷ **Mensaje.** Es el mecanismo por el cual se transmite información entre un emisor y un receptor. Por ejemplo, en una carta, la información está codificada en forma de texto en un papel; esta información codificada, junto con el sobre en que es enviada es lo que conforman el mensaje.
- ▷ **Metadatos.** En ciertos lenguajes, los metadatos son información que permite estructurar, describir o caracterizar ciertos datos. Visto de otra forma, los metadatos son "datos acerca de los datos" o "información sobre la información".
- ▷ **Microsoft.NET.** Es la plataforma de desarrollo de Servicios Web implementada por Microsoft Corporation.
- ▷ **Middleware.** Como middleware se conoce a todo el software que utilizamos como apoyo entre las capas dentro de un sistema, como pueden ser directorios LDAP, aplicaciones de Bases de Datos, Software de RAID, Contenedores de aplicaciones, etc.
- ▷ **MIDlet.** Se refiere a cualquier aplicación Java que cumpla con el estándar MIDP de J2ME y puede ser ejecutada en dispositivos portátiles.
- ▷ **MIDP.** Mobile Information Device Profile. Es un perfil que extiende la configuración CLDC, para soportar dispositivos que tengan una mayor capacidad de despliegue de interfaces gráficas.

- ▷ **Mochila, Problema de la.** Es el nombre dado a un problema matemático. Básicamente, teniendo un grupo finito de elementos conocido A y sea y un elemento de A , se busca una forma eficiente de encontrar un subconjunto de elementos x_i de A tal que $x_1+x_2+x_3+\dots+x_n=y$.
- ▷ **Montículo.** Es un área de memoria que utiliza el Recolector de Basura de Java para almacenar objetos referenciados y candidatos a recolección.
- ▷ **MSIL.** Microsoft Intermediate Language. Lenguaje Intermedio Microsoft. Es un lenguaje arquitecturalmente neutro desarrollado por Microsoft para su uso por el CLR de .NET. Es la contraparte del *bytecode* de Java.
- ▷ **MTU.** Maximum Transfer Unit. En redes ethernet, se refiere al tamaño máximo que tendrá un paquete para ser transmitido en un solo acceso al canal.
- ▷ **Multiplexión.** Es un método mediante el cual es posible enviar varios mensajes a través de un canal compartido de forma simultánea. Un mux en el emisor es encargado de combinar los mensajes para enviarlos por el canal y en el receptor el demux es el encargado de separarlos.
- ▷ **MUX.** Multiplexor. Es un dispositivo que se encarga de combinar, con alguna técnica, varios mensajes para enviarlos simultáneamente por un canal hacia el receptor. CDMA y TDMA son ejemplos de métodos de multiplexión.
- ▷ **NameSpaces.** Ver *Espacios de Nombres*.
- ▷ **NASDAQ.** Se refiere al indicador económico de un conjunto de empresas líderes en el sector de tecnología que cotizan en la bolsa de valores de Nueva York.
- ▷ **NCP.** Network Control Protocol. Es un protocolo estándar obsoleto de comunicación en una red de datos, que se utilizó inicialmente en ARPANET y posteriormente fuera sustituido por TCP/IP
- ▷ **NetBEUI.** NetBIOS Extended User Interface. Es una versión mejorada del protocolo NetBIOS.
- ▷ **NetBIOS.** Network Basic Input Output System. Es un protocolo de redes que permite buscar, reconocer y acceder a dispositivos interconectados en una red de datos.
- ▷ **Nodo (Red).** Es un dispositivo capaz de recibir y enviar información dentro de una red de datos.
- ▷ **NP.** Problema computacional que puede ser resuelto por una máquina de Turing No Determinista (no están definidas todas las transiciones posibles o existe algún estado donde la transición puede ser múltiple).
- ▷ **NTSC.** National Television Systems Committee. Estándar internacional que define la forma de transmitir señales de televisión (30 cuadros de imagen por segundo y 525 líneas de resolución). Es usado en la mayoría de los países.
- ▷ **Nudos Incas.** Era un sistema de contabilidad usado por los incas a base de nudos hechos en una cuerda.

6. Anexos

- ▷ **ODBC.** Open DataBase Connectivity. Es un estándar abierto que proporciona una API consistente para el acceso a la información, independientemente de la fuente de datos de donde provenga.
- ▷ **OMG.** Object Management Group. Es un grupo de compañías de la industria de las tecnologías de información cuya meta es acelerar la adopción de software basado en objetos y componentes a través del establecimiento de recomendaciones, guías y estándares para el desarrollo de aplicaciones.
- ▷ **Orientación a Objetos (Lenguajes, Programación).** Es una forma de diseñar programas y lenguajes de programación, donde todo se puede ver como un objeto con sus propios atributos y comportamiento. Como ejemplos de éstos lenguajes tenemos: Java, C++, C#, SmallTalk. También existen lenguajes que no son totalmente orientados a objetos, como Perl, Pascal, BASIC, etc.
- ▷ **Ortogonal, Conjunto de Instrucciones.** En un conjunto de instrucciones de alguna máquina en particular, la ortogonalidad implica que cualquier combinación posible de cualquier característica del lenguaje tiene significado para la máquina, facilitando a su vez su implementación.
- ▷ **OSI (Modelo).** Open Systems Interconnection. Es un modelo estándar de referencia que sirve para diseñar sistemas de comunicación modulares, robustos y extensibles, proclamado por la ISO.
- ▷ **P2P (Peer to Peer).** Se refiere a que la comunicación entre 2 sistemas es directa, y no mediante un servidor intermedio que controle la conexión o el flujo de información.
- ▷ **P2P (Redes).** Redes Peer to Peer. Método de conexión en red directa entre usuarios a través de Internet sin necesidad que exista un servidor intermedio. Su finalidad es facilitar y optimizar el intercambio de información entre usuarios, pero también se les usa para la búsqueda e intercambio de material ilegal y/o piratería.
- ▷ **PAL.** Phase Alternation Fine. Estándar internacional que define la forma de transmitir señales de televisión (25 cuadros de imagen por segundo y 625 líneas de resolución). Es usado principalmente en países europeos.
- ▷ **Palm.** Palm Incorporated. Una de las compañías más importantes de desarrollo de hardware (PalmOne Inc.) y software (PalmSource Inc.) de PDA's, con sede en Sunnyvale, California. También se refiere coloquialmente a algún dispositivo PDA.
- ▷ **PCS.** Personal Communication Services. Es un estándar que define la prestación de servicios de comunicación de telefonía celular.
- ▷ **PDA.** Personal Digital Assistant. En lenguaje cotidiano, generalmente se refiere a algún dispositivo portátil que nos provea de capacidad de lectura, almacenamiento y procesamiento de información.
- ▷ **Probador de Teoremas.** Es una máquina que, en base a ciertas reglas y criterios, puede decidir si otra máquina satisface una condición o cumple con determinadas características.

- ▷ **Problema de la Mochila.** Ver *Mochila, Problema de la*.
- ▷ **Punto de Acceso.** En redes, es un punto por el cual un dispositivo se puede conectar y acceder a la red local, y puede ser alámbrico (un socket) o inalámbrico (una antena).
- ▷ **Rabin-Williams (Criptografía).** Técnica de cifrado de llave pública basado en el problema del logaritmo discreto.
- ▷ **RAID.** Redundant Array of Inexpensive Disks. Un RAID es una forma de interconectar varios discos de forma que puedan ser vistos por un sistema como una sola unidad de disco de mayor capacidad. Esta técnica (que puede ser implementada por software o por hardware) se usa principalmente para contar una solución de espacio de almacenamiento de bajo costo, de mayor desempeño, de alta disponibilidad y/o que garantice al máximo la integridad de los datos. Existen diferentes niveles y esquemas de RAID, dependiendo de las necesidades que se deban satisfacer (por ejemplo: RAID 0, RAID 1, RAID 3, RAID 5, RAID 1+0, RAID 3+1, RAID 5+1)
- ▷ **Ratio (Display).** Se refiere a la relación entre el ancho y el alto de un dispositivo de visualización gráfica (display). Un ratio de salida de 1:1 denomina a una pantalla exactamente cuadrada (el tamaño de alto y ancho es el mismo). Existen otros ratios de salida muy comunes: por ejemplo, las televisiones son más anchas que altas, (una tv convencional tienen un ratio de salida de 4:3 y en una panorámica es de 16:9).
- ▷ **RDBMS.** Relational DataBase Management System. Sistema de Manejo de Bases de Datos Relacionales. Se denomina RDBMS a cualquier software que maneje bases de datos por el modelo entidad-relación, como por ejemplo: MySQL, PostgreSQL, Oracle, MS-SQL Server, Access, FoxPro, etc.
- ▷ **Recolector de Basura.** El recolector de basura es un programa que corre dentro de cualquier máquina virtual Java, y está encargado de liberar la memoria que no es utilizada dentro de una aplicación. Mediante algún algoritmo específico, el recolector de basura está encargado de averiguar qué localidad(es) de memoria ya no son referenciables (y por tanto inalcanzables) dentro de la aplicación, para entonces marcar esas porciones de memoria como disponibles para su uso.
- ▷ **Repetidoras.** En telecomunicaciones, cualquier canal de comunicación sufre algún tipo de atenuación (es decir: la potencia de una señal transmitida irá disminuyendo en función de la distancia que viaja a través del canal). Para evitar la pérdida de información en la señal a causa de la atenuación, las repetidoras tienen la tarea de recibir una señal atenuada, amplificarla y volverla a inyectar al canal de transmisión. La distancia que una señal puede viajar sin perder información, es función directa de las características del canal de comunicación y el medio de transmisión, y esto a su vez determina la necesidad de colocar o no repetidoras, así como la distancia máxima que puede existir entre una y otra
- ▷ **Retorno de Inversión.** Se refiere al periodo de tiempo en que una tecnología permite recuperar el dinero invertido en su compra e implementación.

6. Anexos

- ▷ **Retransmisión.** Un dispositivo o mecanismo de retransmisión, tiene la función de recibir mensajes desde una fuente y retransmitirlos sin modificación alguna a un destino diferente. Esto sucede cuando por alguna razón es imposible que el emisor contacte directamente al destinatario del mensaje. Por ejemplo: en la comunicación con dispositivos inalámbricos, muchas veces por limitaciones geográficas, económicas o técnicas, se utiliza un satélite como retransmisor entre la fuente de la información y el dispositivo de destino.
- ▷ **Roaming.** Del inglés "vagar", actualmente se aplica a las redes de telefonía celular, y es cuando un usuario de determinado proveedor de servicios, sale de su área de cobertura originalmente asignada, pero aún fuera de esa área puede usar la señal proporcionada por otra red, ya sea de su propio proveedor, de un tercero, o una red pública para continuar haciendo y recibiendo llamadas. En ocasiones este servicio puede ser solicitado explícitamente, pero ya se comienza a generalizar el uso de Roaming Automático. Este concepto se puede extender a cualquier otro tipo de red inalámbrica cuando un dispositivo es capaz de cambiar dinámicamente el punto de acceso que utiliza para conectarse a una red sin perder la señal.
- ▷ **ROI.** Return Of Investmen. Ver *Retorno de Inversión*.
- ▷ **RPC.** Remote Procedure Call. Es un modelo de programación que nos permite hacer que un sistema pueda invocar métodos que residen en algún otro sistema remoto.
- ▷ **RSA (Criptografía).** Rivest, Shamir and Adelman. Técnica de Cifrado de llave pública. El Acrónimo RSA está formado por las iniciales de los apellidos de los desarrolladores.
- ▷ **RTE.** Run Time Environment. Es el conjunto de recursos que un programa tiene para ser ejecutado (arquitectura, memoria, pila, procesador, almacenamiento secundario, etc.)
- ▷ **Ruido.** En comunicaciones, son señales que existen naturalmente en el ambiente, y se suman a otras señales, distorsionándolas. Por lo general, el ruido consta de señales aleatorias, de frecuencia aleatoria y de baja potencia.
- ▷ **Sandbox.** Ver *Caja de Arena*.
- ▷ **Server-side.** En un ambiente cliente-servidor es cuando una aplicación o un proceso es ejecutado en el procesador del servidor y los resultados pueden o no ser enviados al cliente.
- ▷ **Servidor de Aplicaciones.** Ver *Contenedor de Aplicaciones*.
- ▷ **Servlet.** Tecnología Java que permite invocar aplicaciones que residen en algún servidor remoto, desplegadas a través de un servidor de aplicaciones.
- ▷ **SGML.** Standard Generalized Markup Language. En realidad, SGML es un metalenguaje proclamado estándar en 1986 que sirve para definir otros lenguajes de tipo Markup (o de marcas). Como ejemplos de lenguajes creados a partir de SGML, tenemos HTML y XML.

- ▷ **Simplex.** Modo de comunicación unidireccional. Es decir: un emisor únicamente puede enviar mensajes en una sola dirección al receptor, pero nunca en sentido contrario.
- ▷ **Single Sign On (SSO).** Es un mecanismo por medio del cual un objeto (ya sea un programa, un dispositivo, una computadora o un usuario) puede mantener una misma identidad a través de varios sistemas heterogéneos sin la necesidad de autenticarse ante cada uno de ellos (esta funcionalidad la proporciona el esquema SSO y es transparente al objeto).
- ▷ **SOAP.** Simple Object Access Protocol. Es un estándar que define la forma en que los Servicios Web pueden comunicarse, enviarse mensajes, invocar métodos y recibir resultados entre sí.
- ▷ **SQL.** Structured Query Language. Lenguaje de Consultas Estructurado. Es un lenguaje estándar promulgado por la ANSI para el acceso a datos almacenados en un RDBMS.
- ▷ **SQLJ.** SQL Java. Estándar del lenguaje SQL usado por cualquier controlador JDBC.
- ▷ **SSL/TSL.** Secure Sockets Layer / Transport Layer Security. Protocolo de cifrado de clave pública (asimétrico) que permite encriptar la comunicación entre 2 sistemas a nivel de transporte.
- ▷ **SunONE.** Es la plataforma de desarrollo de Servicios Web implementada por Sun Microsystems.
- ▷ **Swing.** Versión extendida y mejorada de AWT para crear GUI's.
- ▷ **Tablas Caldeas.** Eran unas planchas de barro que utilizaban los caldeos, donde se acumulaban los conocimientos y en algunas se hacían marcas para realizar contabilidad.
- ▷ **Tablet PC.** PC Tableta. Son dispositivos semiportátiles que brindan funcionalidades similares a las de una PC de escritorio, generalmente del tamaño de un cuaderno o un portafolios.
- ▷ **TCO.** Total Cost of Ownership, Costo Total de Propiedad. Es un término administrativo que se refiere a un estimado del total de gastos que se generan, directa o indirectamente, del hecho de poseer y/o implementar una determinada plataforma de tecnología de información.
- ▷ **TCP/IP.** Transmission Control Protocol / Internet Protocol (Protocolo de control de transmisión / Protocolo de Internet). Es el protocolo estándar más utilizado para la comunicación entre 2 sistemas a través de Internet. Actualmente el más usado es el TCP versión 4 (TCPv4), pero se está comenzando a usar el protocolo estándar que usará Internet 2: TCP versión 6 (TCPv6).

6. Anexos

- ▷ **TDMA.** Time Division Multiple Access. Tecnología de multiplexión utilizada por el sistema de telefonía celular GSM. La forma de transmitir múltiples mensajes de forma aparentemente simultánea, es utilizar una banda por intervalos específicos de tiempo, es decir: si deseamos transmitir n mensajes simultáneamente, lo que se haría con TDMA, es dedicar un periodo de tiempo a la transmisión de un paquete correspondiente al primer mensaje, el siguiente periodo de tiempo se usa para transmitir el segundo mensaje, el siguiente para el tercero, y así sucesivamente. Después de que se ha transmitido un paquete del mensaje n , el siguiente periodo de tiempo se transmite el segundo paquete del mensaje 1. El ciclo se repite hasta terminar de transmitir todos los mensajes.
- ▷ **Thread.** En Java se conoce como Thread a un hilo de ejecución. Un hilo de ejecución tiene su propio ambiente de ejecución, lo cual permite que pueda ser ejecutado independientemente de otros hilos. Las provisiones que proporciona Java para el manejo de Threads, le permiten realizar una o varias tareas (o programas, o procesos) simultáneamente.
- ▷ **Timeout.** En comunicaciones, se refiere al periodo de tiempo que un emisor puede esperar por la respuesta del receptor a un mensaje (es decir: un tiempo de caducidad de la conexión). Si se cumple este periodo de tiempo sin que el emisor obtenga respuesta, se puede asumir un error en la comunicación. Cuando la comunicación es síncrona, este mecanismo permite desbloquear la aplicación del emisor y seguir con su flujo de ejecución normal (con la posible aparición de un error derivado del error en la comunicación).
- ▷ **Token Ring.** Es una arquitectura de redes en donde la topología es de anillo, y existe un único paquete de datos especial llamado *token* (o *estafeta*). El dispositivo que tenga el *token* en un momento determinado, es el único que tiene la capacidad de comunicarse con otro dispositivo dentro de la red.
- ▷ **Topología de Red.** Se refiere a la forma en que están conectados un determinado número de dispositivos en una red de datos. Estas pueden ser: bus (todos los dispositivos conectados a 1 solo cable por donde viajan los datos), estrella (todos los dispositivos conectados aun dispositivo central), árbol (dispositivos organizados en forma jerárquica) o anillo (un dispositivo conectado directamente a otro, y éste a su vez a otro, hasta formar un anillo cerrado).
- ▷ **Touchscreen.** Pantalla sensible al tacto. Dispositivo de entrada de datos en un computadora integrado con un dispositivo de visualización. Generalmente consta de un monitor o una pantalla plana sensibles al tacto. Los elementos gráficos de la pantalla responden cuando se toca dentro de cada uno de ellos con el dedo o con un lápiz apuntador. Se utiliza generalmente como reemplazo de un mouse.
- ▷ **Transposición.** Se denomina transposición al intercambio en el orden de los elementos de algún conjunto. En Criptografía, la transposición se usa para lograr la difusión de la información, protegiendo un mensaje cifrado contra el criptoanálisis por algoritmos basados en la frecuencia de repetición de ciertos caracteres o palabras en un lenguaje específico.
- ▷ **TSL.** Transport Layer Security. Es un método de establecer conexiones seguras entre 2 sistemas. Ver *SSL*.

- ▷ **Tuneleo.** Es una técnica mediante la cual un mensaje codificado dentro de un protocolo, se "encapsula" o se "envuelve" dentro de un mensaje en un protocolo diferente. Eso se usa generalmente para compatibilidad en las comunicaciones entre emisor y receptor o para proveer mecanismos de cifrado a protocolos que por sí mismos no los implementan. Como ejemplo de esto último, tenemos el protocolo HTTPS.
- ▷ **Tunneling.** Ver *Tuneleo*.
- ▷ **UDDI.** Universal Discovery, Description and Integration specification. Es una especificación estándar que define la forma en que los Servicios Web pueden buscarse, encontrarse y describirse entre sí de forma dinámica.
- ▷ **UDP.** User Datagram Protocol. Protocolo de Datagramas de Usuario. Protocolo de comunicación en redes de datos basado en datagramas. Ver *Datagrama*.
- ▷ **URL.** Uniform Resource Locator. Localizador Uniforme de Recursos. ;ecanismo que permite nombrar de manera consistente los objetos y/o recursos en Internet. Una URL consta típicamente de un protocolo de comunicación, un puerto (a veces implícitamente asociado al protocolo), la dirección de un servidor y la dirección de un recurso en específico. También se le conoce coloquialmente como *dirección de internet*.
- ▷ **USB.** Universal Serial Bus. Estándar de comunicaciones alámbricas entre dispositivos y computadoras desarrollado por Apple Computer Inc., permite conexiones en cascada (un solo puerto USB puede comunicarse hasta con 127 dispositivos conectados en serie) a velocidades de transmisión de hasta 12Mbps para USB 1.0 y de 480Mbps para USB 2.0. Se utiliza principalmente para conectar dispositivos (como teclados, ratones impresoras, escáners o cámaras de video y fotográficas) a computadoras.
- ▷ **Virus (Informática).** Es un programa de computadoras bautizado con este nombre debido a sus características similares a los virus biológicos. Los virus de computadoras generalmente infectan un sistema sin conocimiento del usuario, pueden o no dañarlo y pueden o no autoreplicarse para infectar otras computadoras y extender la epidemia y el daño; algunos virus tienen la finalidad de presentar propaganda comercial al usuario, o simplemente molestarlo. Existen vacunas (o programas antivirus) encargados de identificar los programas de virus en una computadora y eliminarlos del sistema.
- ▷ **W3C (WWW Consortium).** Consorcio WWW. Es un organismo internacional encargado de promulgar estándares sobre las tecnologías utilizadas en la WWW.
- ▷ **WAP.** Wireless Application Protocol. Es un protocolo que permite el intercambio de información entre una red de datos y dispositivos portátiles, como celulares y PDA's a través de una aplicación conocida como navegador WAP.
- ▷ **WCDMA.** Wideband CDMA. Tecnología de telecomunicaciones de tercera generación, CDMA de banda ancha, que es mucho más rápido que CDMA. Su multiplexión se basa en el mismo principio que CDMA, pero soporta velocidades de hasta 2Mbps para la transmisión voz, datos, video e imágenes. Fue adoptada como estándar por la ITU bajo el nombre "IMT-2000" (International Mobile Communications 2000).

6. Anexos

- ▷ **Web Service.** Es un estándar que nos permite diseñar, implementar y publicar aplicaciones modulares mediante el uso de lenguajes y protocolos estándares como XML, UDDI, SOAP, WSDL, HTTP y HTTPS, y que a su vez, nos permite construir sistemas muy complejos a partir de dichos módulos, independientemente de que estén en sistemas, arquitecturas e inclusive lenguajes de programación diferentes.
- ▷ **Wi-Fi.** Wireless Fidelity. Ver 802.11.
- ▷ **WLAN.** Wireless LAN. Area de Red Local implementada con canales de transmisión inalámbricos.
- ▷ **WORA.** Ver *Write Once, Run Anywhere*.
- ▷ **Write Once, Run Anywhere.** Escriba una vez, Ejecute en cualquier lugar. Característica que se atribuye a los programas Java debido a la independencia de plataforma de su bytecode.
- ▷ **WS.** Ver *Web Service*.
- ▷ **WSDL.** Web Services Description Language. Lenguaje en que se definen las funciones y comportamiento específico de un Web Service para que puedan ser publicados en algún Contenedor de Aplicaciones.
- ▷ **WUSB.** Wireless USB. Es un grupo de trabajo formado por varias empresas líderes en el sector de telecomunicaciones encargado de desarrollar y promover el estándar WUSB, que será básicamente una conexión USB por un medio de transmisión inalámbrico. Se podrán transmitir datos a velocidades de hasta 480Mbps a una distancia máxima de 10m. Su ventaja sobre su competencia directa (bluetooth) es su velocidad de transmisión, lo cual a su vez implica su desventaja: un mayor consumo de energía.
- ▷ **WWW.** World Wide Web. Red Mundial Ancha. Se denominar coloquialmente WWW a la red de servidores http que existen en Internet.
- ▷ **XHTML.** XML-HTML. Es un lenguaje cuya funcionalidad es exactamente la misma que HTML, pero sigue una sintaxis mucho más estricta. XHTML fué creado con XML con la intención de eliminar algunos malos hábitos de los programadores de páginas que han deformado el estándar HTML (creado con SGML).
- ▷ **XML.** eXtensible Markup Language. XML es un lenguaje que sirve para crear lenguajes igual que SGML (en realidad, XML es un subconjunto de SGML, sólo que mucho más sencillo de entender e implementar). XML ha servido para implementar muchos otros lenguajes estándares en la última década (SOAP, UDDI, WSDL, XHTML, etc.), aunque puede también ser utilizado para generar un lenguaje completamente nuevo y personalizado que se adecue a necesidades específicas.

*** BIBLIOGRAFÍA ****

- ▷ *A Mathematical Theory Of Communication*. C.E. Shannon, W. Weaver. Universidad de Illinois. 1949.
- ▷ *Arquitectura de la máquina virtual Java*. Revista Digital Universitaria. <http://www.revista.unam.mx/vol.1/num2/art4/>
- ▷ *Asegurando Servicios Web*. Sitio de XML.com. <http://webservices.xml.com/pub/a/ws/2003/01/15/ends.html>
- ▷ *Características de Java*. Instituto Tecnológico de Apizaco. <http://www.itapizaco.edu.mx/paginas/JavaTut/froufe/parte2/cap2-5.html>
- ▷ *Criptografía*. El Rincón de Quevedo. <http://rinconquevedo.iespana.es/rinconquevedo/Criptografia/introduccion.htm>
- ▷ *Especial de Comunicaciones Móviles, 2a. parte*. Monografías.com. <http://www.monografias.com/especiales/comunicamov/index2.shtm>
- ▷ *Estándares de Telecomunicaciones*. Página Personal de Evelio Martínez Mtz. <http://www.eveliux.com/articulos/stds.html>
- ▷ *Exploring Wireless LAN's*. Del Mar College, Corpus Christi, Texas, E.U. <http://www.delmar.edu/Courses/ITNW2313/wireless.htm>
- ▷ *Historia de Internet*. SAE Consulting. http://www.saeconsulting.com/ciberflat/3_1.htm
- ▷ *Historia de las Computadoras (Asistentes Personales)*. Página del Museo de la Informática y Computación Aplicada. <http://mx.geocities.com/pcmuseo/computadoras/historia.htm>
- ▷ *Historia de las Telecomunicaciones (COFETEL)*. http://www.cft.gob.mx/inf_telecom_laera.html
- ▷ *Introducción a las Redes de Datos*. Universidad de Los Ángeles, Puebla (nodo TAU: Asociación civil) <http://www.tau.org.ar/base/lara.pue.udlap.mx/redes/rede196.htm>
- ▷ *J2EE vs. Microsoft.NET*. TheServerSide Java Enterprise Community. <http://www.theserverside.com/articles/article.tss?l=J2EE-vs-DOTNET>
- ▷ *Java: El lenguaje de Internet*. Professional Help Desktop Consultores. <http://www.phd.es/pdf/JAVA2R.pdf>
- ▷ *MIDP GUI Programming*. Sun Developer's Site <http://developers.sun.com/techttopics/mobility/midp/articles/ui/>
- ▷ *Network Software*. School of Computing and Information Technology at the Nathan campus of Griffith University, Brisbane, Australia. <http://www.cit.gu.edu.au/teaching/3142CIT/slides/ch1/node1.html>
- ▷ *Palm Bluetooth Guide*. PalmZone.net <http://www.palmzone.net/modules.php?name=Sections&artid=2>

** Todos los sitios se encuentran en servicio al día 07/May/2004.

6. Anexos

- ▷ *Persistent Storage Using MIDP RMS*. Sun Developer's Site
<http://developers.sun.com/techttopics/mobility/midp/articles/ui/>
- ▷ *Telecomunicaciones, Las*. Departamento de Ingeniería de Sistemas, Universidad de los Andes.
<http://agamenon.uniandes.edu.co/museo/indice.htm>
- ▷ *Wireless Developer Home Page*. Sun Microsystems.
<http://developers.sun.com/techttopics/mobility/>

ENLACES DE INTERÉS ^{††}

- ▷ Apple Computer Inc.
<http://www.apple.com>
- ▷ Bluetooth Official Membership Page.
<http://www.bluetooth.org/>
- ▷ Bluetooth Wireless Official info site
<http://www.bluetooth.com/>
- ▷ Comisión Federal de Telecomunicaciones (COFETEL).
<http://www.cofetel.gob.mx/>
<http://www.cft.gob.mx/>
- ▷ Especificaciones de SOAP.
<http://www.w3.org/TR/soap/>
- ▷ Especificaciones de WSDL.
<http://www.w3.org/TR/wsdl/>
- ▷ Evans Data Corporation
<http://www.evansdata.com>
- ▷ Gartner Research Inc.
<http://www.gartner.com>
- ▷ IBM Corporation.
<http://www.ibm.com>
- ▷ IEEE.
<http://www.ieee.org>
- ▷ IETF.
<http://www.ietf.org>
- ▷ Infrared Data Association.
<http://www.irda.org/>
- ▷ Internet 2 Home Page
<http://www.internet2.org>
- ▷ ISO OnLine.
<http://www.iso.org>
- ▷ ITU.
<http://www.uti.org>
- ▷ J2EE Home Page.
<http://java.sun.com/j2ee/index.jsp>
- ▷ J2ME Home Page.
<http://java.sun.com/j2me/index.jsp>
- ▷ J2SE Home Page.
<http://java.sun.com/j2se/index.jsp>
- ▷ Java Card Technology Home Page.
<http://java.sun.com/products/javacard/index.jsp>

^{††} Todas las ligas están actualizadas y funcionando al día 07/May/2004.

6. Anexos

- ▷ Java Home Page (Sun).
<http://java.sun.com/>
- ▷ Java Tutorials. Sun Microsystems
<http://java.sun.com/developer/onlineTraining/>
- ▷ JDeveloper. Oracle Corp.
<http://otn.oracle.com/products/jdev/>
- ▷ kSOAP Open Source Project. The Enhydra Project.
<http://ksoap.enhydra.org/>
- ▷ kXML Open Source Project. The Enhydra Project.
<http://kxml.enhydra.org/>
- ▷ Microsoft Corporation.
<http://www.microsoft.com/>
- ▷ Microsoft.NET Home Page. Microsoft Corporation.
<http://www.microsoft.com/net/>
- ▷ Multiband OFDM Alliance Home Page
<http://www.multibandofdm.org/>
- ▷ Object Management Group
<http://www.omg.org/>
- ▷ Oracle Technology Network. Oracle Corp.
<http://otn.oracle.com>
- ▷ Palm Inc.
<http://www.palm.com>
- ▷ PalmOne Inc.
<http://www.palmone.com>
- ▷ PalmSource Inc. (PalmOS Home Page).
<http://www.palmsource.com>
- ▷ RSA Security Inc.
<http://www.rsasecurity.com>
- ▷ Sony Corp.
<http://www.sony.com>
- ▷ Sun Microsystems.
<http://www.sun.com/>
- ▷ SunOne Home Page.
<http://www.sun.com/software/sunone/>
- ▷ Symbian OS Home Page.
<http://www.symbian.com/>
- ▷ UDDI Home Page.
<http://www.uddi.org/>
- ▷ Web Services Activity Community.
<http://www.w3.org/2002/ws/>
- ▷ Web Services Interoperability Organization.
<http://www.ws-i.org/>

- ▷ Wi-Fi Alliance Official Site.
<http://www.weca.net/OpenSection/index.asp>
- ▷ WiMedia Alliance.
<http://www.wimedia.org/>
- ▷ Windows Mobile Home Page
<http://www.microsoft.com/windowsmobile/products/pocketpc/default.mspx>
- ▷ World Wide Web Consortium (W3C).
<http://www.w3c.org>
- ▷ Xerox Company.
<http://www.xerox.com>
- ▷ XML Home Page.
<http://www.xml.org>